uc3m | Universidad Carlos III de Madrid

Grado en Ingeniería de Sistemas Audiovisuales
Curso 2016/2017

Trabajo Fin de Grado

“Despliegue de un sistema de IPTV en un emulador de redes”

Anabel Ortiz Alegre

Tutor

Ignacio Soto Campos

Presentación 18 de Octubre de 2017

Agradecer a mis padres por su amor incondicional,
su apoyo y sus consejos de los que tanto he
aprendido.
A mi hermano por escucharme cuando más lo necesitaba.
Este TFG se lo dedico a ellos.
Agradecer también a mi tutor por su amabilidad,
paciencia y conocimientos enseñados.

TÍTULO: Despliegue de un sistema IPTV en un emulador de redes

AUTOR: Anabel Ortiz Alegre

TUTOR: Ignacio Soto Campos

La defensa del presente Trabajo Fin de Grado se realizó el día 18 de Octubre de 2017, siendo calificada por el siguiente tribunal:

PRESIDENTE: Pablo Zumel Vaquero

SECRETARIO: Antonio Ángel Pastor Vallés

VOCAL: Pablo Martínez Olmos

Habiendo obtenido la siguiente calificación:

CALIFICACIÓN:

Presidentente **Secretario** **Vocal**

Resumen

En las últimas décadas se han producido cambios en la forma de consumir los servicios multimedia, actualmente el desarrollo de dispositivos móviles ha permitido que se pueda disponer de servicios audiovisuales en cualquier lugar. El tráfico de datos móviles mundial ha aumentado cada año dando lugar a la Era del Zettabyte; de este tráfico una gran parte corresponde a vídeo. Dar servicio de calidad a esta creciente demanda es crucial para el éxito de los proveedores de servicios. Por lo que necesitan dimensionar sus redes y probar nuevas funcionalidades y mejoras utilizando herramientas que emulen las características de la red, y así evitar tener que utilizar equipos físicos. IPTV, Televisión sobre el protocolo IP, es un servicio que desde su lanzamiento ha ganado año tras año suscriptores, y necesita estas herramientas para evaluar el rendimiento de sus redes. En este TFG se desarrolla un entorno de pruebas para tráfico de vídeo basado en un emulador de redes, siendo éste su objetivo principal.

Abstract

In the last decades there have been changes in the way of consuming multimedia services, currently the development of mobile devices has made it possible to have audiovisual services anywhere. Global mobile data traffic has increased every year, giving rise to the Zettabyte Era; of this traffic a large part corresponds to video. Giving quality service to this growing demand is crucial to the success of service providers. So they need to scale their networks and test new features and improvements using tools that emulate the characteristics of the network, and thus avoid having to use physical equipment. IPTV, Television over the IP protocol, is a service that since its launch has gained year after year subscribers, and needs these tools to evaluate the performance of their networks. In this TFG is developed a test environment for video traffic based on a network emulator, this being its main objective.

Índice

1.INTRODUCCIÓN……....13
1.1. Motivación……...13
1.2. Objetivos……14
1.3. Estructura……15
2. ESTADO DEL ARTE…..……16
2.1. Introducción……16
2.2. Arquitectura de red……19
2.3. Calidad de servicio (QoS).....21
2.4. Unicast……..……22
2.5. Multicast……..……...23
2.5.1. Protocolos de encaminamiento multicast……....25
2.5.2. Protocolo Multicast Independiente-Modo Disperso...26
2.6. CORE (Common Open Research Emulator).....28
2.7. Proyectos anteriores……29
2.8. Marco regulador……29
3. DISEÑO DE LA SOLUCIÓN……32
3.1. Instalación de software y configuración……32
3.1.1. Quagga……..35
3.1.2. XORP……37
3.1.3. Conectando con la máquina anfitriona……42
3.2. Arquitectura de red……42
3.2.1. Escenarios de red unicast……..44
3.2.2. Escenarios de red multicast……...46
3.3. Aplicación de distribución de vídeo……48
4. RESULTADOS Y EVALUACIÓN……49
4.1. Entrega tráfico unicast……..49

4.1.1. Escenario de red conectado con máquina anfitriona mediante interfaz dummy……...49
4.1.2. Escenario de red con nodo host para servidor……...52
4.1.3. Escenario de red con varios hosts clientes……54
4.2. Entrega tráfico multicast……..54
4.2.1. Escenario de prueba……...54
4.2.2. Escenario final multicast……56
4.3. Casos de estudio en entorno de pruebas……57
4.3.1. De tráfico multicast a unicast……57
4.3.2. Plano de datos……62
4.4. Evaluación de los resultados obtenidos……66
5. ENTORNO SOCIO-ECONÓMICO……..67
5.1. Planificación……...67
5.2. Presupuesto……..68
5.2.1. Costes del personal……68
5.2.2. Costes de los recursos utilizados……..68
5.2.3. Otros costes directos……..69
5.2.4. Costes indirectos……69
5.3. Impacto socio-económico……70
6. CONCLUSIONES……...71
Anexo A: Configuración de VLC……...72
Anexo B: Extended abstract……..78

Índice de figuras

2.1. Arquitectura típica de IPTV……21
3.1. Nodos virtuales en la capa de red……33
3.2. Canvas en CORE con nodos virtuales dibujados..……....34
3.3. Nodos enlazados a través de la herramienta Link……34
3.4. Ventana de CORE……...36
3.5. Servicios disponibles en CORE……36
3.6. Configuración servicio RIP……37
3.7. Configuración servicio OSPF……39
3.8. Configuración servicio PIM-SM……41
3.9. Ventana configuración control de red……...43
3.10. Arquitectura de red IPTV……..43
3.11. Escenario de red con nodo RJ45……44
3.12. Escenario de red con dos hosts como servidor y cliente……....45
3.13. Escenario de red con varios hosts……..46
3.14. Escenario de prueba multicast……47
3.15. Escenario final multicast……...48
4.1. Lanzamiento VLC en el host cliente……50
4.2. Servidor emitiendo flujo de vídeo……..51
4.3. Cliente recibiendo flujo de vídeo……...51
4.4. Lanzamiento VLC en el host servidor……..52
4.5. Lanzamiento VLC en el host cliente……53
4.6. Servidor emitiendo flujo de vídeo……..53
4.7. Cliente recibiendo flujo de vídeo……...54
4.8. Comando iperf como cliente en el nodo host servidor n1……55

4.9. Comando iperf como servidor en el nodo host n2……..56
4.10. Cliente recibiendo flujo de vídeo……57
4.11. Comando iperf cliente en el host servidor para multicast……...58
4.12. Comando iperf cliente en el host servidor para unicast……..59
4.13. Salida de comandos del cliente iperf en el nodo host servidor de unicast……...60
4.14. Salida de comandos de servidor iperf en el host cliente de multicast……...60
4.15. Salida de comandos del servidor iperf en el nodo host cliente de unicast……...61
4.16. Escenario de red CORE durante prueba iperf……..61
4.17. Configuración de RP estático……..63
4.18. Primeros paquetes capturados en el router DR……...64
4.19. Primeros paquetes capturados en el nodo host cliente……..64
4.20. Árbol de distribución con router RP estático……65
4.21. Primeros paquetes capturados en el router DR……...65
4.22. Primeros paquetes capturados en el host cliente……66
5.1. Planificación para realización de TFG……67
5.2. Costes del personal……68
5.3. Costes de recursos……..68
5.4. Costes fungibles……..69
5.5. Coste de transporte……69
5.6. Resumen de costes……69
5.7. Presupuesto final……70
A.1. Interfaz gráfica VLC en el nodo emulado……72
A.2. Emitir……73
A.3. Abrir medio……..74
A.4. Protocolo……..75
A.5. Configuración dirección y puerto……..76
A.6. Configuración TTL……...77
A.7. Configuración recepción……..77

Capítulo 1

INTRODUCCIÓN

1.1. Motivación

A finales de los 80 se realizaron las primeras investigaciones sobre la transmisión de televisión sobre el protocolo IP. Tras varios intentos se logró una transmisión exitosa, y las investigaciones continuaron sirviéndose del desarrollo de nuevas tecnologías y dispositivos. Surge la dificultad de tener que llegar a los abonados con un gran ancho de banda. Esto obligó a los proveedores a dimensionar sus redes para la transmisión de IPTV.

El término IPTV fue introducido por Judith Estrin y Bill Carrico tras la fundación de la empresa Precept Software en 1995. La empresa desarrolló una aplicación software que transmitía tráfico de audio y vídeo de diferentes calidades [SMRT].

En Reino Unido, Kingston Communications introdujo IPTV sobre DSL (Digital Subscriber Line) en 1999, y VoD también sobre DSL en 2001. En Francia, Orange lanzó ADSL en 1999 y los proveedores de servicios franceses lanzaron triple play[1] en 2006 [JGR]. En España, Telefónica lanzó IPTV en 2004 a través de Imagenio.

[1] Triple play es la oferta de mercado de disponer en única factura de los servicios de Internet, telefonía y televisión.

Desde su introducción los servicios de IPTV han experimentado un gran crecimiento, pues permiten una nueva forma de ver la televisión dando al espectador control y personalización de lo que visualizan. En Francia es donde se ha visto una fuerte expansión, siendo las previsiones de suscriptores para 2020 de 18,1 millones, mientras que en España esta cifra será de 4,4 millones [Subs]. Esta diferencia se debe a la gran demanda que hay en Francia gracias al nivel de competencia del mercado y los precios asequibles.

Cisco en su informe Visual Networking Index (VNI) [VNI] de este año proporcionaba el dato de que el tráfico de vídeo sumaba un 60% del tráfico de datos total en el mundo. Por otra parte, predicen que más de las tres cuartas partes del tráfico de datos móviles del mundo serán de vídeo para 2021, con un incremento de 9 veces entre 2016 y 2021 llegando al 78% del tráfico total al final del período.

Un parámetro básico en el desarrollo y las pruebas de un sistema de IPTV es la calidad de experiencia (QoE) que mide el rendimiento del servicio para asegurar que se cumplan las expectativas del usuario. Los proveedores necesitan asegurar la calidad de experiencia haciendo especial hincapié en los flujos de vídeo que son sensibles a retardos y variaciones del retardo en la red. El tráfico de vídeo consume muchos recursos de la red, por lo que los proveedores tienen la necesidad de dimensionar y configurar su red de manera que se cubra la creciente demanda. Por lo tanto se necesitan plataformas que permitan evaluar soluciones de red para tráfico IPTV.

1.2. Objetivos

En este trabajo se desarrolla una plataforma de pruebas para tráfico IPTV basada en un emulador de redes. El diseño e implementación de una red no es una tarea liviana; al principio se realizaban pruebas sobre equipos reales para ensayar nuevas funcionalidades con el fin de mejorar el rendimiento de la red. Se crearon los emuladores y simuladores para seguir haciendo pruebas sin tener que utilizar equipos reales de forma que se reducían los costes. Un emulador de redes es una herramienta que permite reproducir las características de una red, de forma que se puede evaluar su funcionamiento. El emulador utilizado en este TFG es CORE (Common Open Research Emulator) que permite emular redes en una o varias máquinas. Además con CORE se puede conectar una red emulada con redes físicas reales, y fue desarrollado por un grupo de investigación que forma parte de la

división de Investigación y Tecnología de Boeing, siendo un proyecto de código abierto.

El objetivo principal de este proyecto es la creación de un entorno de pruebas para tráfico de vídeo basada en CORE, en la que finalmente se realizan diferentes pruebas con el fin de validar dicha plataforma. Para alcanzar este objetivo principal se tiene una serie de objetivos secundarios. Uno de ellos es familiarizarse con el programa CORE y las tecnologías de distribución de vídeo, para lo cual se configura el servicio de tráfico unicast y se prueba la entrega de vídeo. Otro objetivo secundario es configurar el servicio de tráfico multicast en el emulador de redes CORE. También se tiene como objetivo secundario implantar una aplicación de distribución de vídeo basada en VLC. Finalmente, se realizan diferentes pruebas y casos de estudio para validar la plataforma.

1.3. Estructura de la memoria

La memoria se divide en seis capítulos cuyo contenido se describe brevemente a continuación:

- Capítulo 1, Introducción: contiene la motivación y objetivos de este Trabajo Fin de Grado, además de este apartado.
- Capítulo 2, Estado del arte: contiene la explicación de conceptos teóricos sobre IPTV, una descripción de los protocolos de enrutamiento que se usan para la entrega de vídeo en el sistema IPTV, proyectos anteriores relacionados con éste y el marco regulador.
- Capítulo 3, Diseño de la solución: en esta parte se encuentran las diferentes tareas que se han llevado a cabo para la realización del TFG, así como las diferentes topologías implementadas en el emulador.
- Capítulo 4, Resultados: se detallan todos los resultados obtenidos de las pruebas realizadas en cada escenario.
- Capítulo 5, Entorno socio-económico: se detallan la planificación y elaboración del presupuesto de este proyecto, además del impacto socio-económico del proyecto
- Capítulo 6, Conclusiones: aparecen las conclusiones extraídas de este TFG.
- Anexos: aparecen la configuración del framework VLC media player y el resumen extendido en inglés.

Capítulo 2

ESTADO DEL ARTE

2.1. Introducción

El Focus Group on IPTV de la Unión Internacional de Telecomunicaciones (UIT) [5] proporcionó la siguiente definición: “IPTV se define como servicios multimedia, tales como televisión, vídeo, audio, texto, gráficos y datos ofrecidos sobre redes basadas en IP, gestionadas para proveer el nivel requerido de calidad de servicio y experiencia, seguridad, interactividad y confiabilidad”.

IPTV significa Televisión sobre el protocolo IP, y se trata de un sistema de distribución de señales de televisión a través de una red de banda ancha en la que una parte del ancho de banda es reservado con el fin de garantizar la calidad del servicio. Está basado en el vídeo streaming de forma que el contenido se puede visualizar antes de su descarga completa. Además se ofrecen otros servicios complementarios como el grabador de vídeo personal (PVR, Personal Video Recording) en la memoria del dispositivo decodificador que ha evolucionado a la grabación en la nube, la posibilidad de volver a reproducir un programa desde el principio, el vídeo bajo demanda (VoD) o la guía de programación electrónica (EPG).

Este sistema cambia el concepto de entrega de vídeo pues el proveedor no transmite todos los canales esperando a que el usuario se conecte, sino que envía el contenido seleccionado por el usuario cuando se conecte.

IPTV se utiliza para ofrecer servicios que superan las características y funciones de la televisión por cable o por satélite. Los proveedores de servicios eligen la tecnología IP para ofrecer múltiples servicios a través de una sola red. De esta forma, en un sistema de este tipo se puede ofrecer una programación continua de vídeo a miles de clientes simultáneamente [HG09].

Para poder proporcionar este servicio la infraestructura se divide en diferentes partes, que son: adquisición de la señal de vídeo, almacenamiento y servidores de vídeo, distribución de contenido.

Se pueden utilizar diferentes tecnologías de acceso para ofrecer el servicio IPTV, como DSL o fibra óptica. Independientemente de la tecnología que se usa, se comparten unas características básicas de la red IPTV, que se detallan a continuación [WS07]:

- Un flujo continuo de contenidos: el usuario puede seleccionar el flujo que desea ver uniéndose al flujo en progreso. Esto es idéntico a la televisión tradicional, el espectador decide qué canal visualizar, pero no el contenido de éste. Estos contenidos no son creados ni son propiedad del proveedor de IPTV. Una excepción de esta característica es la entrega de contenido bajo demanda, donde el cliente puede seleccionar vídeos almacenados en un servidor y reproducirlos bajo petición.
- Un formato de contenido uniforme: los sistemas IPTV mayormente utilizan sólo un formato de codificación, de forma que se simplifica la gestión del sistema con un diseño uniforme permitiendo un menor grado de mantenimiento. También se simplifica el diseño del decodificador al no tener que soportar diferentes formatos de compresión.
- Una entrega sobre red privada: para que la entrega de vídeo sea exitosa la red IPTV debe ser diseñada cuidadosamente y debe controlarse. El flujo de vídeo debe ser recibido, generalmente comprimido y convertido en paquetes IP, que son entregados al decodificador. Si los paquetes llegan muy pronto deben almacenarse en el decodificador hasta que sean necesarios, pero si llegan tarde puede interrumpirse la reproducción del vídeo. Para solucionar

esto se puede utilizar una memoria buffer en el decodificador, cuyo tamaño debe seleccionarse cuidadosamente con el fin de evitar que se llene causando retrasos o descarte de paquetes.

- Un flujo de vídeo es visto en la televisión a través del Set Top Box (STB): este dispositivo es el encargado de recibir los paquetes, ordenarlos correctamente, decodificar la señal y producir la salida de vídeo al televisor para su visualización. Adicionalmente, es capaz de recibir comandos que el usuario envía mediante el control remoto y enviarlas a la red, y de generar pantallas para comunicarse con el cliente. De esta forma se consigue que el usuario disfrute de interactividad, pues puede elegir el contenido y realizar diversas acciones, con lo que cambia la forma de ver la televisión.

Se aprecian ciertas diferencias entre IPTV y vídeo por Internet, servicios que se siguen confundiendo pues la tecnología que se usa es la misma. La primera de ellas es la naturaleza del contenido; en vídeo por Internet se trata de un contenido en segmentos discretos en el que el usuario puede elegir cuál ver y el orden de visionado, mientras que en IPTV son flujos continuos. Otra diferencia es el contenido que se puede seleccionar; en vídeo por Internet se pueden encontrar millones de archivos, y en IPTV, además de la programación tradicional, el usuario puede elegir contenidos individuales y consumirlos cuando quiera. La tercera diferencia es el acceso al servicio, pues en televisión por Internet se hace uso de Internet público y en IPTV es una red privada IP. Esto permite asegurar una calidad de servicio a los usuarios de IPTV, pero Internet está basado en *best effort,* con lo cual la calidad de la reproducción puede verse afectada o interrumpida. Otra diferencia es el formato de codificación; la televisión por Internet se usan diversidad de estos formatos, mientras que en IPTV mayormente se usa sólo uno o dos a lo sumo, para simplificar el diseño y la decodificación del STB. Por último, el vídeo por Internet se suele consumir en un ordenador o dispositivo móvil, y la forma más habitual de disfrutar IPTV es mediante un televisor a través del decodificador, aunque también se pueden utilizar otros dispositivos.

La entrega del servicio IPTV a cada suscriptor necesita de una red de distribución de alta velocidad. La comunicación en este servicio es bidireccional entre el usuario y la central de datos del proveedor de servicios, y esto no debe afectar a la calidad del vídeo entregado. Un hecho fundamental en el éxito de este servicio es ofrecer suficiente ancho de banda en “la última milla” [GO08]. La evolución de tecnologías de acceso a Internet ha permitido que el usuario goce de un aumento del ancho de banda en su hogar. La línea de abonado digital (DSL, Digital Subscriber Line) realiza

la transmisión a través de los cables de la red telefónica local. Dentro de esta tecnología se encuentra ADSL, ADSL2, ADSL2+, VDSL. Por su parte, la fibra óptica es el medio de transmisión más avanzado, pues es prácticamente inmune a interferencias electromagnéticas, por lo que las pérdidas de señal son mínimas proporcionando un gran ancho de banda a mayores distancias que el cobre.

Con respecto a la codificación los sistemas IPTV usan generalmente MPEG-2, MPEG-4 AVC o VC-1. Suele ser habitual tener que realizar una transcodificación ya que los datos se reciben codificados en un formato específico y son convertidos a otro elegido por el proveedor. DVB creó un estándar denominado DVB-IPTV [9] para la transmisión de servicios multimedia utilizando una red IP. Una vez que los datos se encuentran codificados mediante un estándar, se encapsulan en paquetes RTP (Real-time Transport Protocol) y éstos en datagramas IP.

2.2 Arquitectura de red

Los proveedores pueden ofrecer el servicio IPTV a millones de suscriptores o sólo a unos miles. En cualquier caso, el diseño y el despliegue de la red es crucial para el éxito comercial. En general, las redes IPTV suelen estar diseñadas para un despliegue en fases.

La arquitectura típica de una red IPTV es como la mostrada en la figura 2.1, donde los nombres aquí expuestos pueden variar según el proveedor de servicios. A continuación se describen las funciones de cada instalación [HG09]:

- Extremo de cabecera (SHE, Super Head End): aquí es donde se recibe todo el contenido de programación, se convierte al formato adecuado y se transmite a la oficina de servicio de vídeo (VSO). En primer lugar, se agrega toda la programación, previamente recogida de las fuentes de contenido. Estas señales se obtienen de receptores vía satélite o mediante una red terrestre. Generalmente la entrega se realiza en un formato digital cifrado para que sólo aquel que tenga las claves de descifrado pueda acceder al contenido. Las señales entrantes se deben convertir a un formato concreto para la correcta transmisión en la red IPTV. Con el fin de simplificar se suele utilizar el mismo formato para todos los canales en cuanto a velocidad de bits, formato de codificación y de paquete. Finalmente, las señales de vídeo comprimidas son transportadas a través de una red a la oficina de servicio de

vídeo. Es posible que en esta instalación también se prepare el contenido de VoD para entregarlo a los servidores de VoD de la oficina de servicio de vídeo. El proceso para ello es el mismo que se ha descrito anteriormente, además se debe incluir cierta información en forma de metadatos como la duración del vídeo o la descripción del mismo. Lo mismo ocurre con la publicidad con la excepción de que se entrega a los servidores de contenidos publicitarios. Puede ser que haya un segundo SHE redundante para evitar, en caso de fallo, dejar de dar servicio a los suscriptores.

- Oficina de servicio de vídeo (VSO, Video Service Office): es el encargado de realizar procesamiento y entrega de señales de vídeo para una región geográfica. Este contenido se recibe del propio extremo de cabecera (SHE) y también de fuentes de programación locales, y se transmite hacia la oficina local final (LEO). El contenido local puede llegar en cualquier formato por lo que se necesita convertirlo al formato estándar necesario para la distribución, al igual que se hacía en el extremo de cabecera; puede que esté a una velocidad de bits diferentes o que esté comprimido con otro sistema de codificación, en cuyo caso se debe realizar una transcodificación. Una vez que se tiene el contenido procesado se debe encapsular en paquetes IP que serán enviados a la instalación siguiente, la oficina local final. En esta ubicación se encuentran los servidores de VoD y de anuncios publicitarios, tanto provenientes del extremo de cabecera como de fuentes locales. El servidor de VoD crea un flujo único para el suscriptor interesado en recibirlo. Por otra parte, para poder ofrecer interactividad a los suscriptores, se localizan servidores que ejecutan el software necesario para recopilar y procesar los comandos de cada STB, en los que es necesario un rápido control. También es importante para el proveedor del servicio evitar la visión y la retransmisión no autorizada, por lo que habrá sistemas para verificar que el contenido que se envía a cada STB sea por el que el suscriptor ha pagado y se realizará un proceso de cifrado de dicho contenido.
- Oficina local final (LEO, Local End Office): si se trata de tecnología ADSL, en esta instalación se haya un DSLAM (Digital Subscriber Line Access Multiplexer) que enlaza todo el tráfico de vídeo que debe llegar a cada abonado. También será necesario un splitter que mezcle las señales para hacer llegar también los servicios de telefonía. Los DSLAM más recientes son capaces de proporcionar multidifusión de manera que replica el flujo de vídeo para diferentes abonados que estén viendo simultáneamente el mismo canal.
- Hogar del cliente: nuevamente si se utiliza tecnología DSL aquí es necesario un splitter que separa las señales y así evitar que el teléfono reciba las

señales de alta velocidad destinadas al módem. Este último dispositivo convierte los datos en formatos que los siguientes puedan entender. Por último, está el Set Top Box, que es necesario para el funcionamiento de IPTV. Éste decodifica la señal de vídeo entrante, genera información en pantalla, y recibe comandos del control remoto como el cambio de canal u otras funciones interactivas.

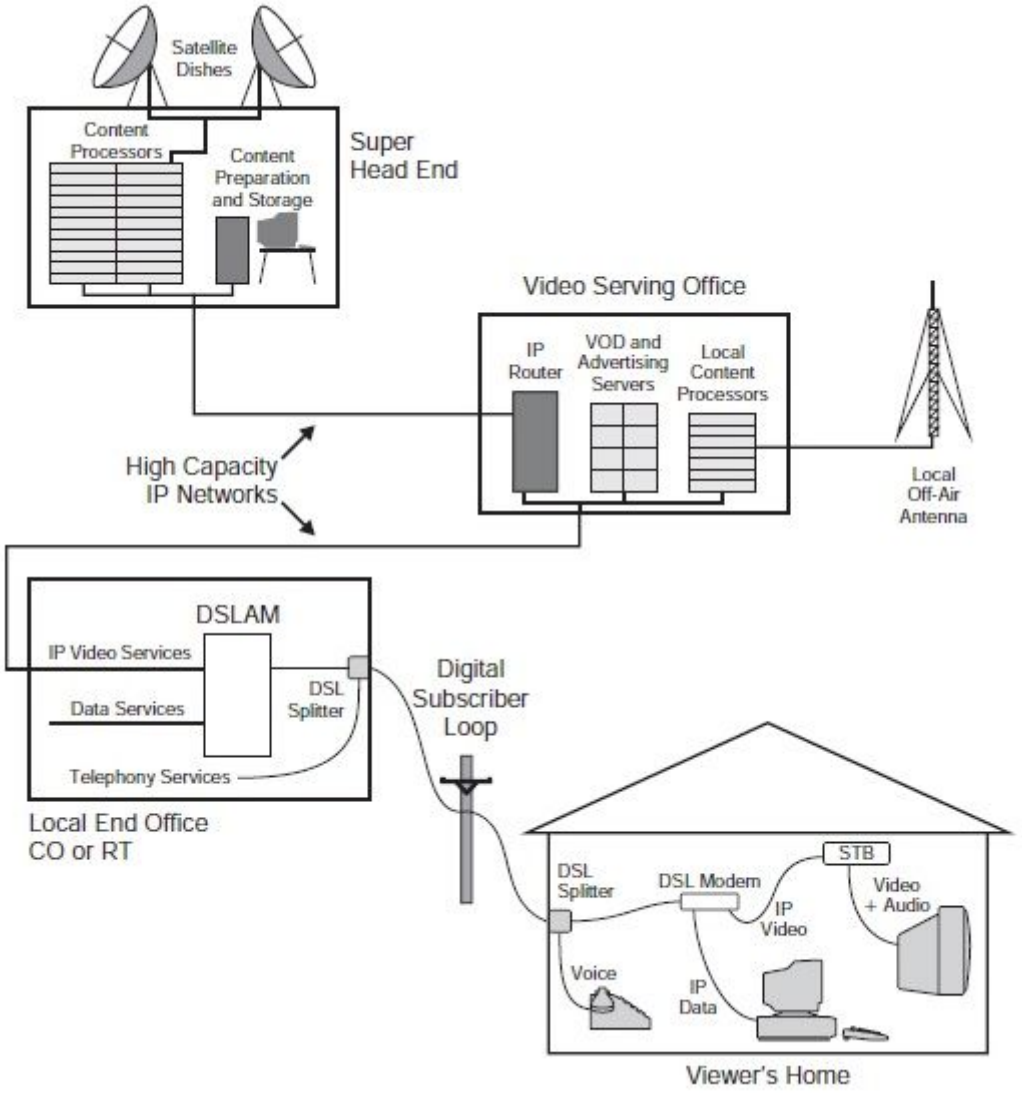

Figura 2.1: Arquitectura típica de IPTV [WS07]

2.3. Calidad de servicio (QoS)

La calidad de servicio depende del ancho de banda de la red de transmisión y del funcionamiento de la codificación. IPTV está basado en transmisiones en tiempo real, por lo que es sensible a fluctuaciones de la red que conllevan a pérdidas de

paquetes o retrasos. Se hace uso de buffers para evitar posibles problemas comentados anteriormente. Es clave dimensionar correctamente el tamaño del buffer para que siempre haya paquetes y se cumplan los requisitos del decodificador [GO08]. El buffer nunca debe saturarse ni estar vacío, en tal caso repercutirá en la calidad del vídeo reproducido. Los parámetros que pueden afectar la calidad de servicio son:

- Jitter y latencia: la variabilidad en la llegada de paquetes y la suma de retardos en una red causa retrasos que conllevan a que el buffer se vacíe y el decodificador no tenga datos para realizar su función. Estas variaciones son causadas por diferentes motivos como problemas de configuración y de servidor.
- Ráfagas: se produce cuando llegan una cantidad de paquetes seguidos al buffer de manera que éste se llena y los paquetes son descartados. Esto se puede controlar mediante el uso de mecanismos de control de admisión.
- Pérdida de paquetes: se puede deber a problemas de software o hardware, a que aumente la velocidad de transmisión de bits. Para garantizar que no se pierdan paquetes se debe priorizar el tráfico de vídeo.

2.4. Unicast

En el modo de transmisión unicast se debe transmitir el flujo de vídeo desde el proveedor de servicio hasta un único receptor. Este método se usa en el servicio de VoD ya que cada suscriptor demandará un contenido diferente al proveedor.

Se hace necesario una tabla de reenvío en cada router de la red para determinar el camino que debe seguir el tráfico hasta el receptor. Para rellenar la tabla de reenvío existen tres modos: en el primero se introducen automáticamente las rutas de las subredes directamente conectadas; en el segundo modo las rutas son introducidas mediante el archivo de configuración del router y se denominan rutas estáticas; por último, están las denominadas rutas dinámicas de forma que la tabla se rellena por un protocolo de enrutamiento.

RIP [RIPv2] es un protocolo IGP (Interior Gateway Protocol) que utiliza el algoritmo de encaminamiento vector-distancia para calcular la mejor métrica de la ruta hacia el destino. Los routers intercambian sus tablas de enrutamiento con sus routers

vecinos, y si un router descubre que una métrica ha cambiado la difunde por broadcast al resto de routers.

OSPF [OSPFv2] también es un protocolo IGP que utiliza el algoritmo de Dijkstra de enlace-estado y el Link State Advertisement (LSA) para describir la topología e información de encaminamiento. Este protocolo divide el dominio en diferentes áreas obteniendo así propiedades de escalabilidad, y dentro de cada área todos los routers deben tener en el mismo Link State Advertisement.

2.5. Multicast

El modo de transmisión multicast entrega el tráfico siguiendo un modelo de uno a muchos o de muchos a muchos. Se hace uso de este tipo de difusión en el servicio IPTV cuando varios suscriptores están visionando el mismo canal simultáneamente. Proporciona un método eficiente de entrega de tráfico ya que se inyecta una menor cantidad de tráfico en la red.

En multicast los emisores envían un flujo de paquetes y la red es la encargada de replicar estos paquetes hasta los receptores que han indicado su interés en recibirlos. Para entregar los datos sólo a los hosts interesados, los routers de la red crean un árbol de distribución, de manera que cada subred con un interesado constituye un rama del árbol. Si hay un nuevo interesado se crea una nueva rama, y si algún receptor deja de estar interesado en recibir los datos su rama es podada.

Los routers configuran el estado de reenvío de multidifusión, que se trata de las interfaces entrantes y salientes de cada grupo de dirección multicast, y lo hacen desde el receptor hasta la raíz del árbol de distribución [BE02]. Es decir, se realiza un reenvío de ruta inversa (RPF, Reverse Path Forwarding) basado en que los routers determinan cuál es la interfaz que está topológicamente más cercana a la raíz del árbol. La tabla de enrutamiento que se utiliza para las comprobaciones RPF puede ser la misma que se utiliza para la difusión unicast o puede ser una tabla dedicada a RPF. En cualquier caso se trata de direcciones unicast ya que las comprobaciones se realizan con estas direcciones. Si se utiliza la misma tabla que con unicast se rellena mediante alguno de los protocolos de enrutamiento comentados en el apartado anterior. Si se utiliza una tabla dedicada de RPF existen protocolos de enrutamiento multicast cuyos mecanismos incluyen rellenar esta tabla como el protocolo DVMRP [DVMRP] (Distance Vector Multicast Routing Protocol).

Otros protocolos como PIM-SM [PIMSM] (Protocol Independent Multicast-Sparse Mode) y PIM-DM [PIMDM] (Protocol Independent Multicast-Dense Mode) hacen uso de otros protocolos para rellenar la tabla.

Cuando un host está interesado en recibir el flujo de datos de una determinada dirección multicast debe comunicarse con el router directamente conectado a él, para ello se utiliza el protocolo de gestión de grupos en Internet (IGMP, Internet Group Management Protocol). El host debe enviar un informe IGMP con el grupo de multidifusión al que quiere unirse a una dirección multicast dedicada; todos los routers de la LAN habilitados para IGMP están escuchando en esa dirección, de forma que cuando escuchan el informe del host uno de ellos utiliza un protocolo de enrutamiento multicast para unirse a ese grupo. Las versiones 1 [IGMPv1] y 2 [IGMPv2] de IGMP sólo permiten especificar la dirección de interés, mientras que con la versión 3 [IGMPv3] se puede especificar también la fuente de emisión. La diferencia entre las versiones 1 y 2 es cómo se realiza el abandono del grupo multicast. En la versión 1 el host deja de enviar informes para ese grupo y el router, pasado un tiempo, asume que ya no está interesado y deja de enviar tráfico de esa dirección. La versión 2 introduce un nuevo mensaje de abandono explícito, de manera que el router cuando lo escucha envía otro mensaje de consulta de ese grupo, con el fin de determinar si hay otros hosts en la red LAN interesados en seguir recibiendo tráfico de esa dirección. Si no hay ningún interesado deja de enviar tráfico inmediatamente.

En el árbol SPT (Shortest Path Tree) la raíz del árbol es la fuente. Desde que un router escucha a un interesado en un grupo multicast intenta unirse al árbol para ese grupo. Para construir el árbol SPT el router realiza una comprobación RPF consultando su tabla de enrutamiento para la dirección de origen. Con la comprobación averigua qué interfaz está más cerca de la fuente, y envía un mensaje de unión por dicha interfaz RPF hacia el siguiente router. El router que recibe el mensaje de unión incluye la interfaz en la que fue recibido a su lista de interfaz saliente para el estado de reenvío de multidifusión, y realiza una comprobación RPF hacia la fuente para enviar el mensaje de unión. Cada router en el camino hacia la fuente repite este proceso hasta que llegue al router directamente conectado a la fuente o llegue a un router que ya tiene el estado de reenvío de multidifusión para ese par fuente-dirección. Así se crea una nueva rama del árbol y pueden enviarse paquetes desde la fuente al receptor.

La raíz de un árbol compartido es un router situado en el núcleo de la red. Un router directamente conectado con un interesado en recibir datos de un grupo multicast intenta unirse al árbol de distribución. Este router no conoce la dirección de la fuente para ese grupo, pero sí sabe que hay un router que conoce esa fuente y todas las demás, la raíz del árbol. Se realiza un proceso para crear el árbol idéntico al realizado para el SPT, con la diferencia de que el mensaje de unión tiene de origen la raíz del árbol. Para enviar el tráfico multicast este router central debe recibir los paquetes desde la fuente, por lo que realiza una comprobación RPF hacia la fuente y envía un mensaje de unión. Una vez que el router central se une al SPT recibe los paquetes de la fuente y los reenvía hacia el receptor.

2.5.1 Protocolos de encaminamiento multicast

Un protocolo se considera protocolo de enrutamiento multicast cuando realiza la configuración del estado de reenvío de multidifusión e intercambia este estado con otros routers habilitados para multicast. Se distinguen principalmente dos categorías: densos y escasos [BE02].

En los protocolos densos se asume que existe una distribución densa de receptores interesados en todo el dominio, es decir, un receptor en cada subred para cada grupo. Estos protocolos siguen un modelo de inundación y poda, en el que se envía tráfico a través del dominio para informar a los routers de las fuentes de multicast. Un router reenvía este tráfico por todas las interfaces excepto por la que recibió los datos, de modo que esa es la interfaz RPF hacia la fuente. Por lo tanto, si se recibe datos de una interfaz que no es la interfaz RPF hacia la fuente se descartan los paquetes y se envía un mensaje de poda. También se envía este mensaje si un router no tiene receptores interesados, es decir, si su lista de interfaz saliente está vacía. Se realiza una inundación periódicamente para actualizar el estado. Este mecanismo permite construir de una manera fácil los árboles de distribución basados en la fuente. Ejemplos de este tipo de protocolo son DVMRP (Distance Vector Multicast Routing Protocol) y PIM-DM (Protocol Independent Multicast-Dense Mode).

DVMRP utiliza el mecanismo estándar de inundación y poda e implementa un protocolo separado para rellenar las tablas de enrutamiento con las que se hacen las comprobaciones RPF. Este protocolo realiza un algoritmo de vector distancia basado en calcular la distancia hasta cualquier nodo de la red, de modo que

requiere que los routers informen a sus vecinos de los cambios de topología periódicamente.

PIM-DM es bastante similar a DVMRP en su modo de funcionamiento, la diferencia es que introduce el concepto de independiente, ya que utiliza una tabla de enrutamiento rellenada mediante cualquier protocolo de enrutamiento unicast, y así realizar las comprobaciones RPF.

Los protocolos escasos hacen la suposición de que hay una distribución escasa de receptores interesados en cada subred. La diferencia entre un protocolo escaso y denso es el modo en que se realiza el descubrimiento de la fuente. El protocolo escaso determina un nodo central que conoce todas las fuentes para cada grupo multicast, siendo éste la raíz del árbol compartido. Este protocolo sigue un modelo de unión explícita, de modo que los datos sólo se envían a los routers que lo han solicitado explícitamente. Cuando un host desea unirse a un grupo multicast, el router directamente conectado se une al árbol de distribución hacia el nodo central. A este nodo le llega el tráfico de la fuente a través del SPT. De esta manera se reduce la cantidad de información de estado de todos los routers que no sean el nodo central y no requiere de inundación para averiguar las fuentes activas; todo lo que necesitan saber es quién es este nodo central y cómo alcanzarlo. La desventaja de este protocolo es que el árbol de distribución desde el nodo central al receptor no será el camino óptimo si el receptor se encuentra más cerca de la fuente. En PIM-SM, se realiza un proceso por el cual el receptor se une al SPT hacia la fuente. Una vez que el router que da servicio a uno o varios receptores en una subred conoce la dirección de la fuente envía un mensaje de unión a esa dirección. Cuando se forma el SPT se reciben los paquetes por duplicado de cada árbol de distribución, por lo que envía un mensaje de poda hacia el nodo central.

Existe un modo denominado denso-escaso que soporta ambos protocolos PIM simultáneamente, que es implementado por Juniper Networks y Cisco Systems. En este modo los grupos considerados densos no se asignan a un nodo central y los paquetes destinados a esos grupos siguen el protocolo PIM-DM, mientras que los grupos escasos se asignan a un nodo central y se ejecuta el protocolo PIM-SM.

2.5.2. Protocolo Multicast Independiente-Modo disperso, PIM-SM

El protocolo PIM-SM es el más extendido para la transmisión multicast. En este protocolo el árbol de distribución es un árbol compartido cuya raíz se conoce como punto de encuentro (RP, Rendezvous-Point). Para que funcione correctamente todos los routers dentro de un dominio deben decidir cuál será el punto de encuentro (RP). Para ello existen tres formas: RP estático, router bootstrap (BSR) y auto-RP patentado por Cisco y compatible con routers Juniper.

El punto de encuentro estático es el método más sencillo y se basa en configurar manualmente en cada router la dirección del RP para cada grupo de multidifusión. Tiene la desventaja de necesitar tener que volver a configurar todos los routers si la dirección del RP cambia. Además, en caso de que el RP no pueda ser alcanzado se requiere de configuración adicional para disponer de otro punto de encuentro de respaldo. Esta limitación se supera con el mecanismo Anycast RP en el que se configuran varios routers con la misma dirección IP en la interfaz de loopback. De esta forma se obtiene redundancia en caso de RP fallido llegando los mensajes al siguiente punto de encuentro configurado con la mejor métrica.

El mecanismo auto-RP se basa en el modo de operación denso para enviar mensajes de control, por lo que los routers que quieran realizar auto-RP deben estar configurados en modo denso-escaso. En este mecanismo, los candidatos a punto de encuentro envían un mensaje de anuncio de RP a la dirección 224.0.1.39 en el que detallan el grupo de direcciones que pueden servir. Los routers configurados como agentes de asignación escuchan en esa dirección los mensajes de anuncio y deciden siguiendo una serie de criterios el punto de encuentro. Una vez decidido el RP para cada grupo lo anuncian con mensajes de asignación a la dirección 224.0.1.40 de forma que el resto de routers aprenden la dirección del RP uniéndose a ese grupo.

PIM bootstrap es un mecanismo muy similar a auto-RP en el que uno o más routers deben seleccionarse como candidatos BSR, para lo cual deben configurarse con una prioridad BSR mayor que cero. Cada candidato BSR envía mensajes bootstrap en todas sus interfaces, de modo que los routers cuando reciben el mensaje procesan el paquete y lo reenvían en todas sus interfaces menos por la que llegó el mensaje. Los routers sólo aceptan los mensajes bootstrap recibidos en la interfaz RPF para la dirección del candidato BSR. Si un candidato BSR recibe un mensaje bootstrap con una prioridad mayor que la suya deja de anunciarse como candidato BSR, por lo que llega un momento en el que sólo hay un candidato BSR. Los candidatos a RP envían mensajes de anuncio de candidato a RP a la dirección BSR

con el grupo de direcciones a las que pueden servir. Esta información será añadida a los mensajes bootstrap, de manera que cada router ejecuta un algoritmo hash para determinar cuál será el punto de encuentro.

Los routers directamente conectados a una fuente reconocen los paquetes de multicast examinando las direcciones de origen y destino, y el TTL (Time To Live), de forma que saben que son los routers designados (DR) o routers de primer salto. Cuando una fuente comienza a transmitir el router designado encapsula los datos en mensajes de registro PIM y envía estos mensajes al punto de encuentro RP mediante enrutamiento unicast. El punto de encuentro puede enviar el paquete encapsulado por el árbol RPT hacia los receptores o enviar un mensaje de unión hacia la fuente para crear el árbol de distribución SPT. Cuando se configura el árbol SPT, el punto de encuentro empieza a recibir paquetes multicast duplicados. Una vez le llega el primer paquete multicast nativo manda un mensaje de registro al router designado para que pare de enviarle mensajes de registro con los datos encapsulados.

2.6. CORE (Common Open Research Emulator)

CORE [CORE] es una herramienta en la que se pueden crear redes virtuales para ejecutar aplicaciones y protocolos. CORE permite conectar el escenario de red con redes reales. El demonio CORE gestiona las sesiones de emulación, y construye redes a partir de la virtualización del kernel para los nodos y del uso de un puente para las redes. El demonio es controlado por la interfaz gráfica (CORE GUI), y ambos se comunican mediante una API basada en sockets conocida como la CORE API. CORE se puede ejecutar en sistemas Linux y FreeBSD. En Linux, que es el sistema elegido en este trabajo, CORE usa la virtualización de espacio de nombres de red Linux para crear nodos virtuales y los une a redes virtuales utilizando Linux Ethernet bridging.

Los emuladores permiten reproducir las características de una red, de manera que se obtienen resultados más fiables de lo que realmente pasa en la red al probar nuevas funcionalidades y mejoras con estas herramientas.

Una alternativa a este emulador es EMANE (Extendable Mobile Ad-Hoc Network Emulator), un framework para modelar redes móviles en tiempo real. Ambos emuladores se pueden integrar, de forma que es posible ejecutar CORE con

EMANE. CORE realiza una emulación por encima de la Capa 3, mientras que EMANE emula las Capas 1 y 2.

2.7. Proyectos anteriores

Existen proyectos realizados previamente en los que se implementa un entorno de pruebas para sistemas IPTV en un laboratorio físico. Un proyecto se basa en implementar una red que consta de un par de clientes conectados a un DSLAM mediante dos routers ADSL, un servidor de vídeo externo que también funciona como router de salida hacia Internet. Una vez configurado todos los equipos se realizan pruebas de conectividad y envío de tráfico. En el TFG presente también se realizan estas pruebas con los diferencia de que la red está desarrollada en un emulador de redes.

Otro proyecto está basado en la implementación en el laboratorio de un sistema de IPTV que incluye elementos de red como sistema de tarificación o contenido bajo demanda. Esta plataforma está construida a partir de diferentes proyectos de software libre y realiza pruebas sobre tráfico de vídeo y los elementos de la red configurados. En este TFG no se implementan estos elementos de red pues no se dispone de virtualización de nodos en el emulador de redes, sólo se centra en el tráfico de flujo de vídeo en un sistema IPTV.

Otro proyecto utiliza el simulador GNS3 para evaluar el funcionamiento de la herramienta cuando se conecta a routers físicos. Este proyecto se realiza por completo en sesiones de emulación en las que no hay conexión con redes físicas reales.

2.8. Marco regulador

Con la introducción de los servicios IPTV en el mercado, la mayoría de países aplicaron las regulaciones de radiodifusión televisiva. Introdujeron marcos reguladores horizontales a la radiodifusión de servicios audiovisuales que aplican regulaciones mínimas, de manera que nuevos servicios como IPTV no están sujetos a las regulaciones ex ante.

En 1989 la Unión Europea aprobó la Directiva 89/552/CEE denominada “Televisión sin fronteras” en la que se disponía de unas medidas básicas respecto a las emisiones, se buscaba promocionar la distribución de obras europeas y se regulaba la publicidad. Esta directiva se revisó en 1997 por última vez y afectaba sólo a los emisores que tenían la responsabilidad editorial de la composición de los programas y los transmitían. Un organismo de radiodifusión sujeto a la jurisdicción de un Estado miembro debía cumplir las normas de ese Estado miembro, y todos los demás Estados miembros deben garantizar la recepción gratuita de sus emisiones.

En 2007 se publicó la Directiva 2007/65/CE denominada “Servicios de medios audiovisuales” que revisa y modifica la anterior Directiva, donde se distinguen dos tipos de medios; lineal y no lineal, y se aplican un conjunto de obligaciones mínimas a los servicios audiovisuales a petición que no contemplaba “Televisión sin fronteras”. Estas obligaciones se refieren a normas armonizadas en materia de protección de menores, dignidad humana y prohibición de la publicidad encubierta. El servicio de VoD no es considerado radiodifusión por su naturaleza de comunicación bidireccional, por lo que no está sometido a estas regulaciones.

En España se promulgaron la Ley 25/1994 por la que se incorpora al ordenamiento jurídico español la Directiva 89/552/CEE, y la Ley 7/2010, cuya última modificación fue en 2015, que incluye el derecho del prestador del servicio de comunicación audiovisual a la autorregulación.

Estas normativas afectan a los proveedores de IPTV, pero no directamente a este trabajo; lo que sí concierne es la estandarización de los diferentes protocolos de encaminamiento.

Ingenieros realizan memorandos conocidos como RFC (Request for Comments) en los que se describe el funcionamiento de protocolos; cada RFC es analizado por la IETF (Internet Engineering Task Force), de forma que regula y publica en Internet los estándares con el fin de que la red funcione cada vez mejor.

Uno de los protocolos configurados en este TFG es RIP (Routing Information Protocol). Existen tres versiones del protocolo; la primera [RIPv1] se diseñó para redes de tamaño pequeño, en las que el máximo número de saltos era 15; en la versión 2 [RIPv2] se diseñó para cubrir aspectos que no se consideraban en la anterior como las interacciones entre sistemas autónomos y protocolos IGP/EGP, las subredes o la autenticación, pero mantenía la limitación de 15 saltos para

asegurar compatibilidad entre ambas versiones; por último, la tercera versión [RIPng] especifica el protocolo de encaminamiento para direcciones IPv6.

Otro protocolo que se configura es OSPF (Open Shortest Path First), que también tiene tres versiones, aunque la primera está obsoleta y ya no es implementada. La versión dos [OSPFv2] es utilizada en redes grandes en las que todos los routers ejecutan el mismo algoritmo en paralelo, construyendo árboles con la ruta más corta siendo éstos la raíz de los mismos. La tercera versión [OSPFv3] describe las modificaciones del protocolo para soportar direcciones IPv6.

Por último se configura PIM-SM (Protocol Independent Multicast-Sparse Mode) cuyo último estándar revisado se publicó en Marzo de 2016. En esta revisión [PIM-SM] se eliminan erratas del estándar previo (RFC 4601) y se eliminan características como el estado (*,*,RP), PIM Multicast Border Router y la autenticación mediante IPsec.

Capítulo 3

DISEÑO DE LA SOLUCIÓN

En este capítulo se describen las tareas que se han realizado para implementar un entorno de pruebas para tráfico de vídeo, objetivo principal de este TFG. Se dispone del emulador de redes CORE, herramienta en la que se desarrolla toda la plataforma, de manera que se definen varias topologías de red para los diferentes escenarios emulados. Se deben configurar los protocolos de encaminamiento en los routers que constituyen estos escenarios para tráfico unicast y multicast. Con el fin de poder validar el entorno se utiliza una aplicación de distribución y reproducción de vídeo, que está basada en el framework VLC media player.

3.1. Instalación de software y configuración

La instalación de CORE requiere ciertos requisitos como el sistema operativo y ciertas características de hardware. Se puede instalar mediante paquetes o código fuente. Ambos recursos se encuentran en la página web de CORE y los pasos a seguir aparecen en el manual del emulador. También se dispone en la página web de una máquina virtual con CORE ya instalado. En este TFG se utiliza una máquina virtual VirtualBox con el sistema operativo Ubuntu 14.04 y se instala CORE con paquetes, y simplemente ejecutando el siguiente comando en el terminal al tratarse de esa versión de sistema:

```
sudo apt-get install core-network
```

Una vez instalado, para poder empezar a utilizar el emulador se debe iniciar el demonio CORE como root con el siguiente comando:

```
sudo /etc/init.d/core-daemon start
```

Y, finalmente ejecutar la CORE GUI como usuario normal:

```
core-gui
```

CORE se puede utilizar mediante la interfaz gráfica o scripts en el lenguaje Python. Lo habitual es usar la interfaz gráfica para dibujar nodos y dispositivos de red en el canvas, pero un script de Python también puede configurar e instanciar nodos y redes importando el módulo CORE Python. Programar en Python está fuera del alcance de este proyecto, por lo que la solución se implementa mediante CORE GUI. Para crear redes cableadas se pueden elegir los diferentes nodos de la barra de herramientas vertical en el modo de edición y unirlos mediante la herramienta Link creando un enlace en el que automáticamente aparecen las interfaces. Estos pasos se muestran en las siguientes figuras 3.1, 3.2 y 3.3:

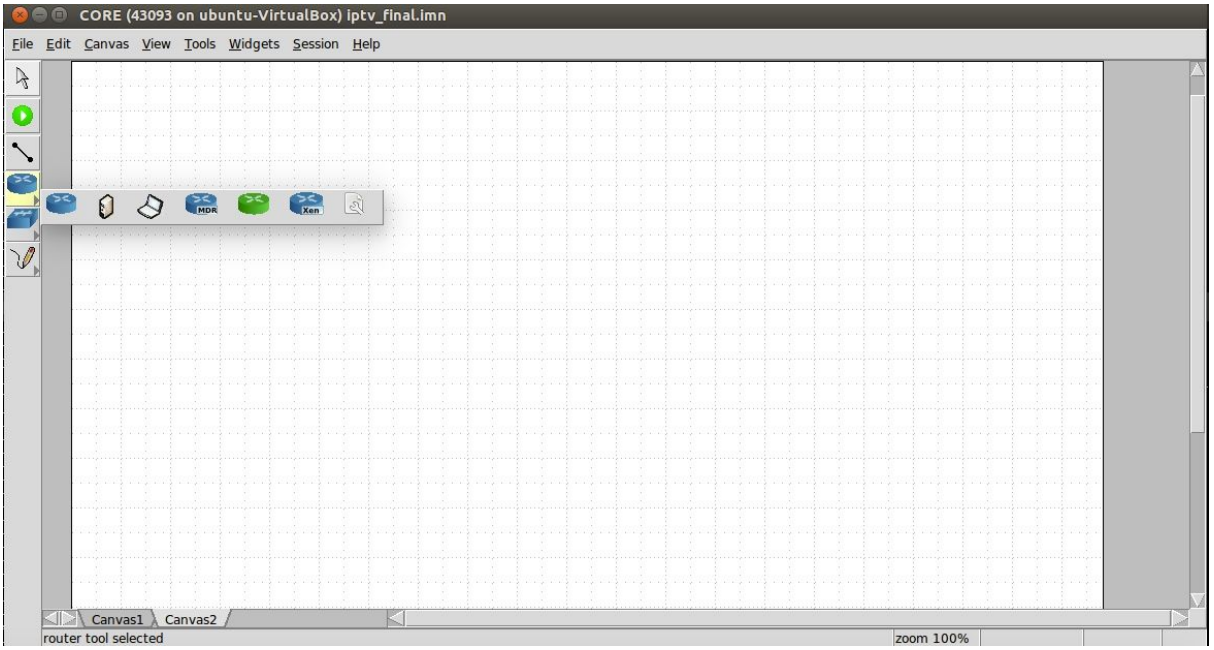

Figura 3.1: Nodos virtuales de la capa de red

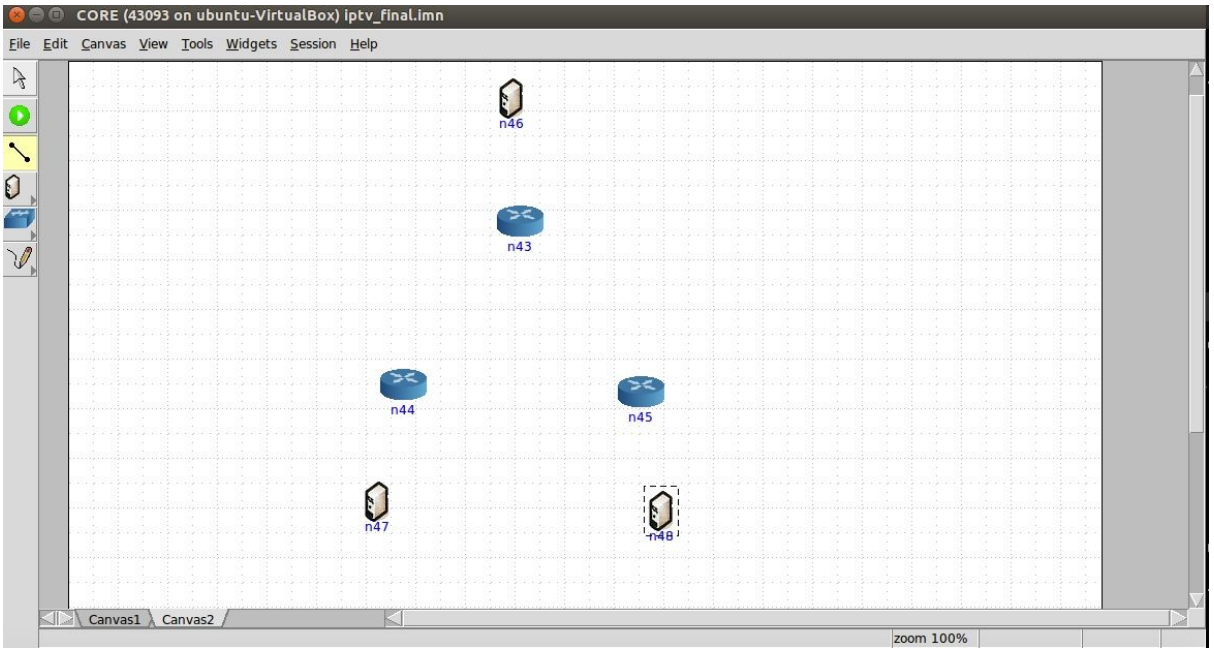

Figura 3.2: Canvas en CORE con nodos virtuales dibujados

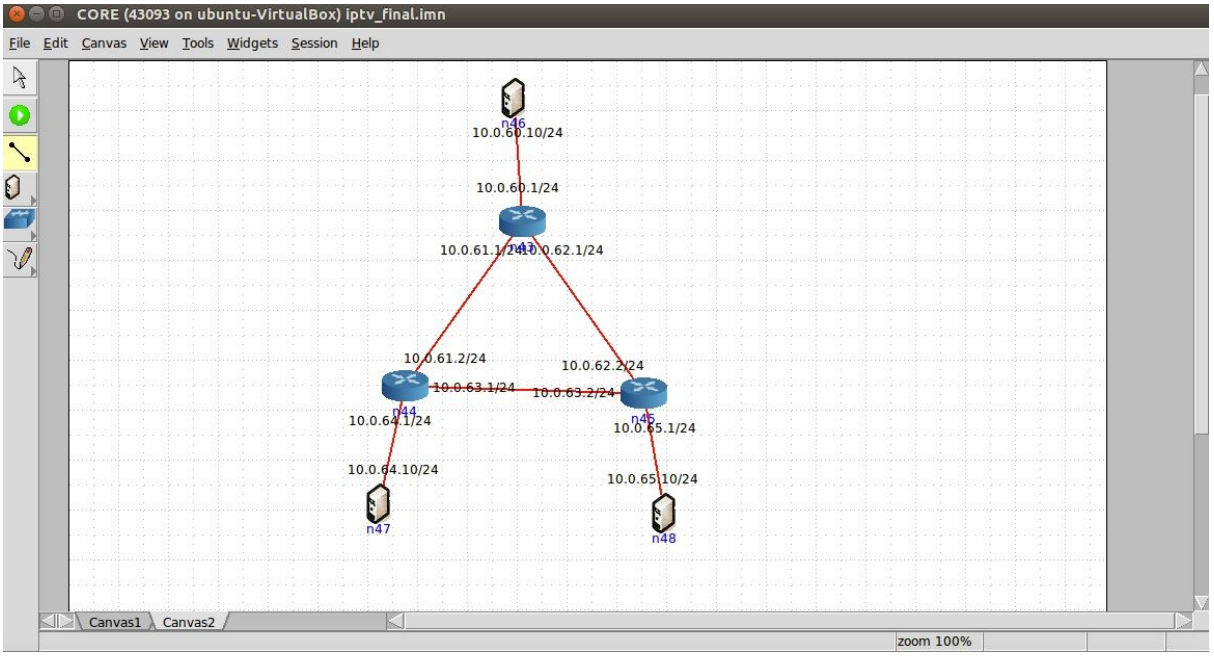

Figura 3.3: Nodos enlazados a través de la herramienta Link

Los servicios se usan en CORE para especificar qué procesos o scripts se ejecutan cuando comienza la emulación. Se pueden personalizar para cada nodo o crear nuevos servicios. En cada servicio se define los directorios por nodo, los archivos de configuración, los comandos de inicio, los comandos de validación, los comandos de apagado y los metadatos asociados a un nodo. En las figuras 3.4 y 3.5 se muestran las interfaces de un nodo y los servicios disponibles.

3.1.1. Quagga

Quagga [QGG] es un suite de software libre de enrutamiento que proporciona implementaciones de protocolos como OSPF, RIP y BGP para plataformas Unix, en particular Linux, FreeBSD, Solaris y NetBSD. Quagga está formado por un demonio denominado zebra que actúa como una capa de abstracción para el núcleo Unix subyacente. Quagga se instala ejecutando el siguiente comando en el terminal:

```
sudo apt-get install quagga
```

Con el objetivo de familiarizarse con el emulador se desarrolla una arquitectura de red para tráfico unicast en la que se configura el servicio RIP (Routing Information Protocol) de Quagga, la misma configuración se utiliza en los equipos reales. Para ello, cada nodo debe tener el servicio RIP configurado y personalizado, de forma que existe un archivo de configuración para cada uno. En el archivo de configuración se declaran las interfaces con sus direcciones IP correspondientes utilizando los comandos `interface` e `ip address`, y también se definen las redes que deben ser anunciadas por el protocolo, utilizando el comando `router rip` para habilitar el protocolo y el comando `network` acompañado de la interfaz para habilitarla y que sea anunciada al resto de nodos. El directorio en cada nodo debe ser `/var/run/quagga`, y los comandos de inicio, parada y validación son los que vienen por defecto. También debe estar habilitado el servicio zebra que está configurado por defecto. La configuración de RIP es la mostrada en la figura 3.6.

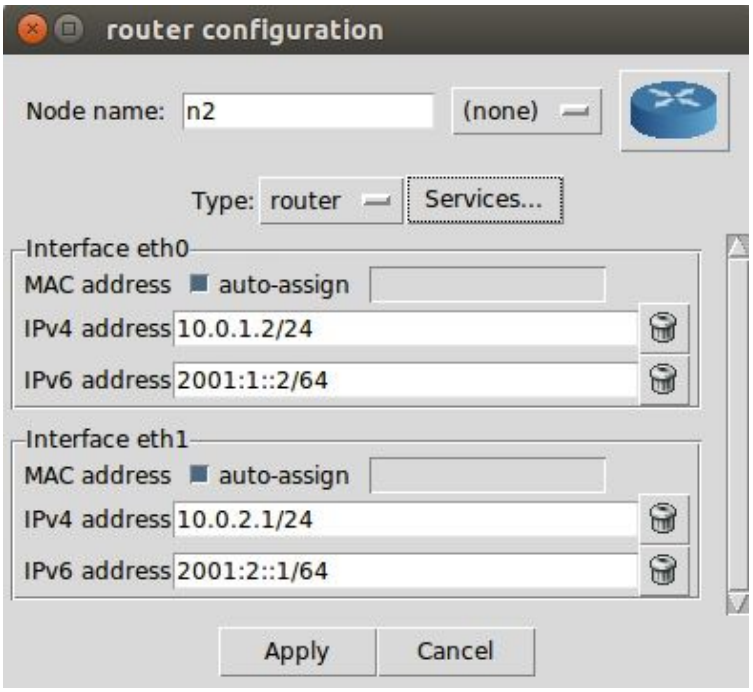

Figura 3.4: Ventana de CORE

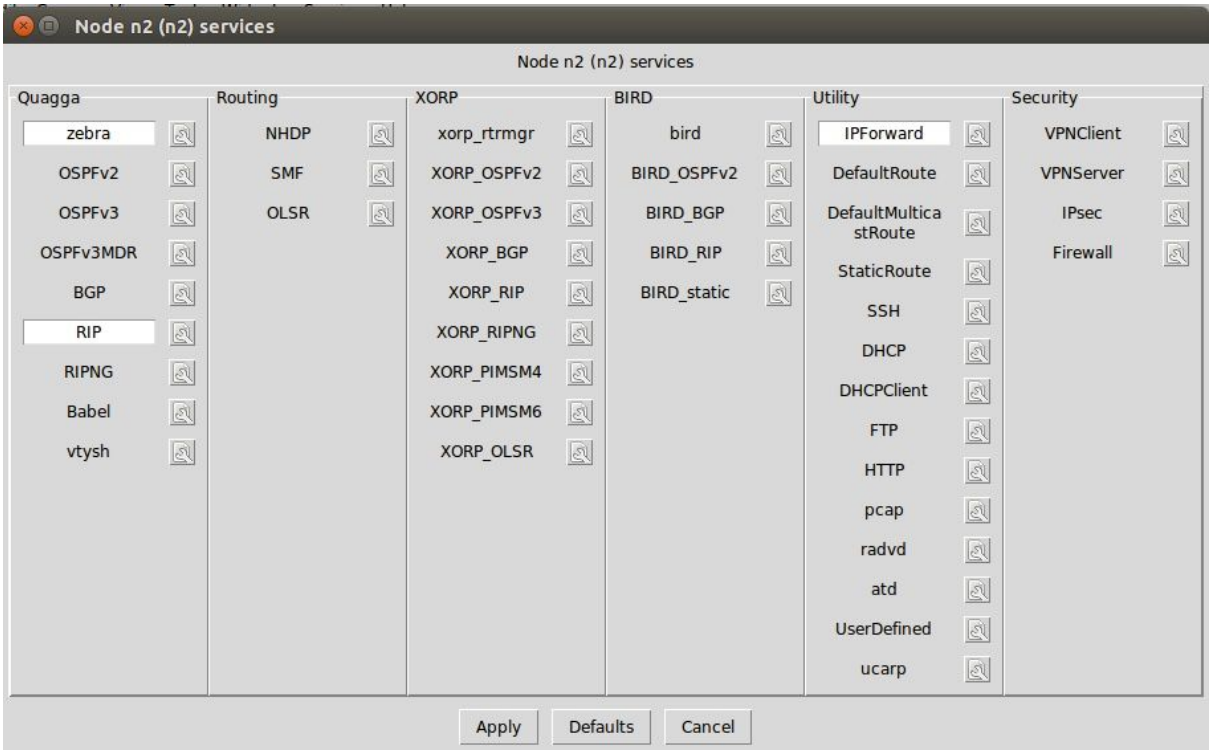

Figura 3.5: Servicios disponibles en CORE

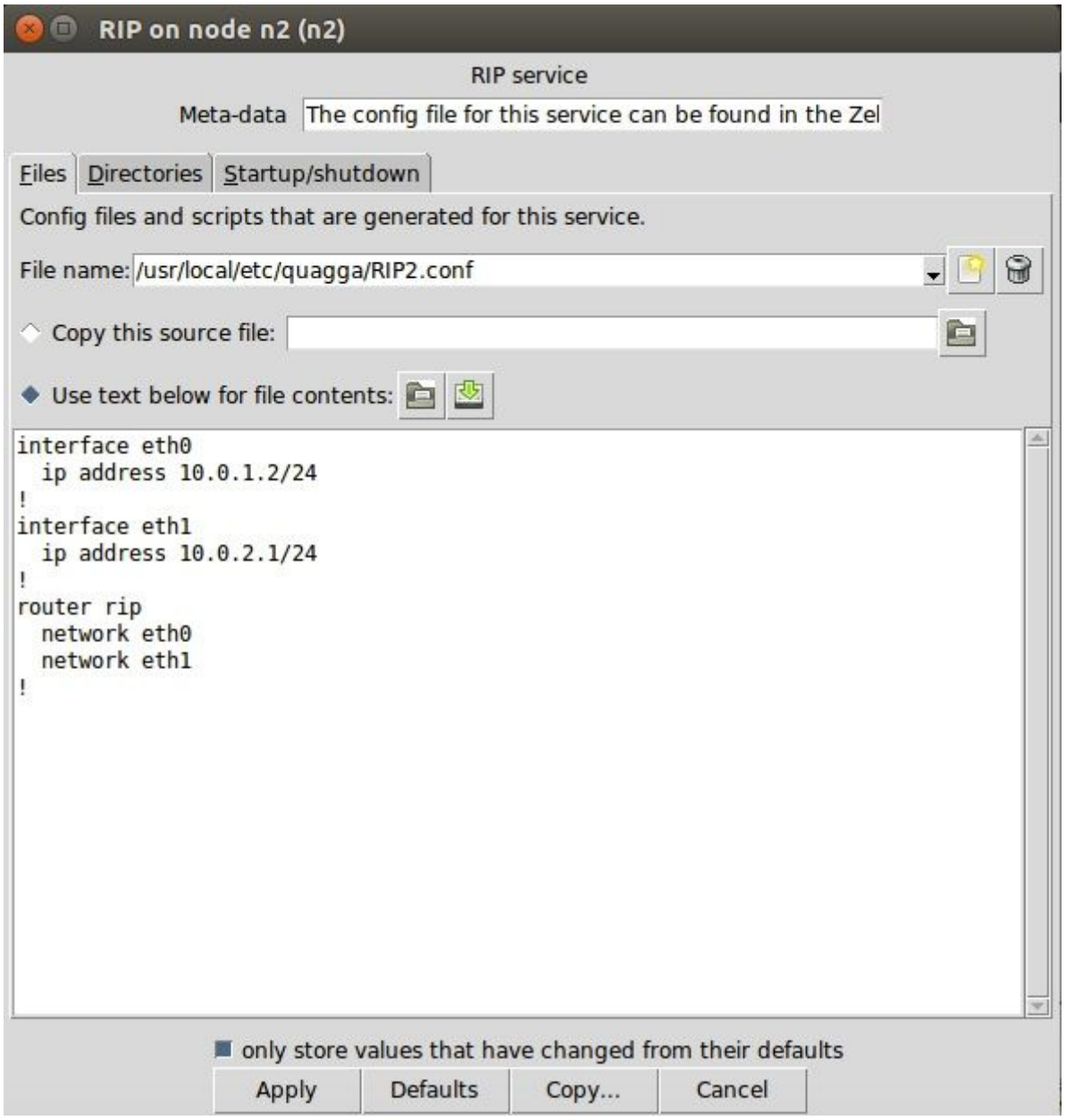

Figura 3.6: Configuración servicio RIP

3.1.2. XORP

XORP [XORP] es una plataforma que proporciona implementaciones de protocolos de enrutamiento, además de ser la única de código libre que permite la configuración de multidifusión. Para la instalación[2] de XORP primero se deben obtener todas las dependencias:

[2] En un principio se realizó la instalación mediante la ejecución de sudo apt-get install xorp, lo que conllevó problemas a la hora de emular los escenarios, que se solucionaron realizando la instalación correctamente.

```
apt-get install build-essential
apt-get install git
apt-get install scons
apt-get install libboost-all-dev
apt-get install libssl-dev
apt-get install libncurses5-dev
apt-get install libpcap-dev
apt-get install traceroute
```

Después obtener la última versión de XORP disponible en Github ejecutando los siguientes comandos:

```
git clone git://github.com/greearb/xorp.ct.git
cd xorp.ct/xorp
```

Por último, se realiza una compilación con el comando scons, y la instalación con el siguiente comando:

```
sudo scons install
```

Para el correcto funcionamiento se deben crear dos enlaces en el directorio /usr/local/bin a los ejecutables xorp_rtrmgr y xorpsh que se instalan en el directorio /usr/local/xorp/sbin. Para ello se ejecutan los siguientes comandos:

```
ln -s /usr/local/xorp/sbin/xorp_rtrmgr /usr/local/bin/xorp_rtrmgr
ln -s /usr/local/xorp/sbin/xorp_rtrmgr /usr/local/bin/xorp_rtrmgr
```

La configuración del tráfico multicast se realiza mediante el servicio OSPF (Open Shortest Path First) para el enrutamiento unicast y PIM-SM (Protocol Independent Multicast-Sparse Mode) para el enrutamiento multicast de XORP.

La configuración del servicio OSPF [Figura 3.7] se basa en definir las interfaces del router especificando la dirección IP con la directiva address dentro de la parte de interfaz virtual vif, que a su vez está dentro de la parte interface. Estas directivas se deben declarar dentro de la directiva area, que delimita un área en el que se configuran las interfaces y los enlaces virtuales de la red. Se debe definir la ID del router mediante la directiva router-id con la dirección IP más pequeña de las interfaces que pertenecen a un router. Este parámetro se utiliza para identificar el dispositivo que origina o procesa información del protocolo.

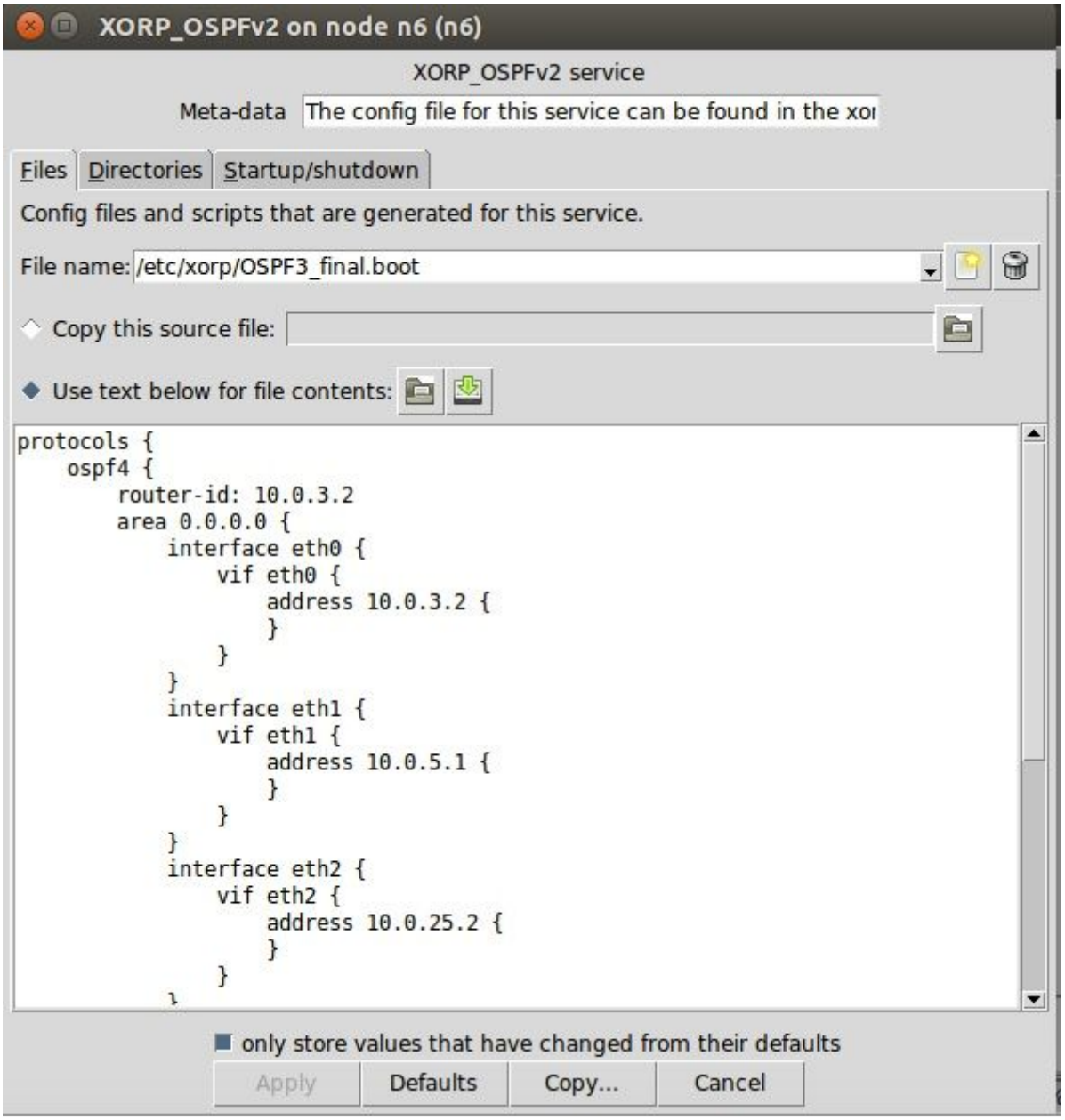

Figura 3.7: Configuración servicio OSPF

La configuración del servicio PIM-SM [Figura 3.8] es algo más compleja incluyendo más directivas que se detallan a continuación:

- interface: se declaran la interfaz que será usada por el protocolo PIM-SM, siendo su valor el nombre que se le ha dado a la interfaz en el archivo de configuración config.boot.
    - vif: especifica la interfaz virtual en la correspondiente interfaz que será usada en el protocolo multicast. El valor de este parámetro debe ser el mismo nombre que tiene en la configuración de las interfaces en config.boot
        - dr-priority: esta directiva toma un valor no negativo que indica la prioridad de la interfaz para convertirse en router designado (DR). Su valor por defecto es 1.

      También debe definirse una interfaz lógica especial con su correspondiente interfaz virtual, ambas con el nombre register_vif, para que el router sea capaz de enviar mensajes de registro al punto de encuentro (RP).

- bootstrap: esta directiva delimita la parte de la configuración en la que se configura el mecanismo bootstrap para determinar el punto de encuentro.
    - cand-bsr: indica que el router es candidato a convertirse en router BSR.
        - scope-zone: especifica el grupo de direcciones multicast a las que está dispuesto ser router BSR. Su valor debe ser una dirección multicast y un prefijo.
        - cand-bsr-by-vif-name: especifica el nombre de la interfaz virtual cuya dirección IP se utilizará para los mensajes bootstrap

    - cand-rp:
        - group-prefix: especifica el rango de direcciones multicast a las que está dispuesto servir como punto de encuentro. Su valor debe ser una dirección multicast y un prefijo.
        - cand-bsr-by-vif-name: especifica el nombre de la interfaz virtual cuya dirección IP se utilizará como la dirección del punto de encuentro.

En los routers directamente conectados con los hosts no se intercambian mensajes correspondientes a los protocolos de enrutamiento, por lo que esas interfaces son

deshabilitadas en la configuración de estos routers. Aún así esas subredes deben ser anunciadas por los protocolos de enrutamiento por lo que es necesario el uso de una policy.

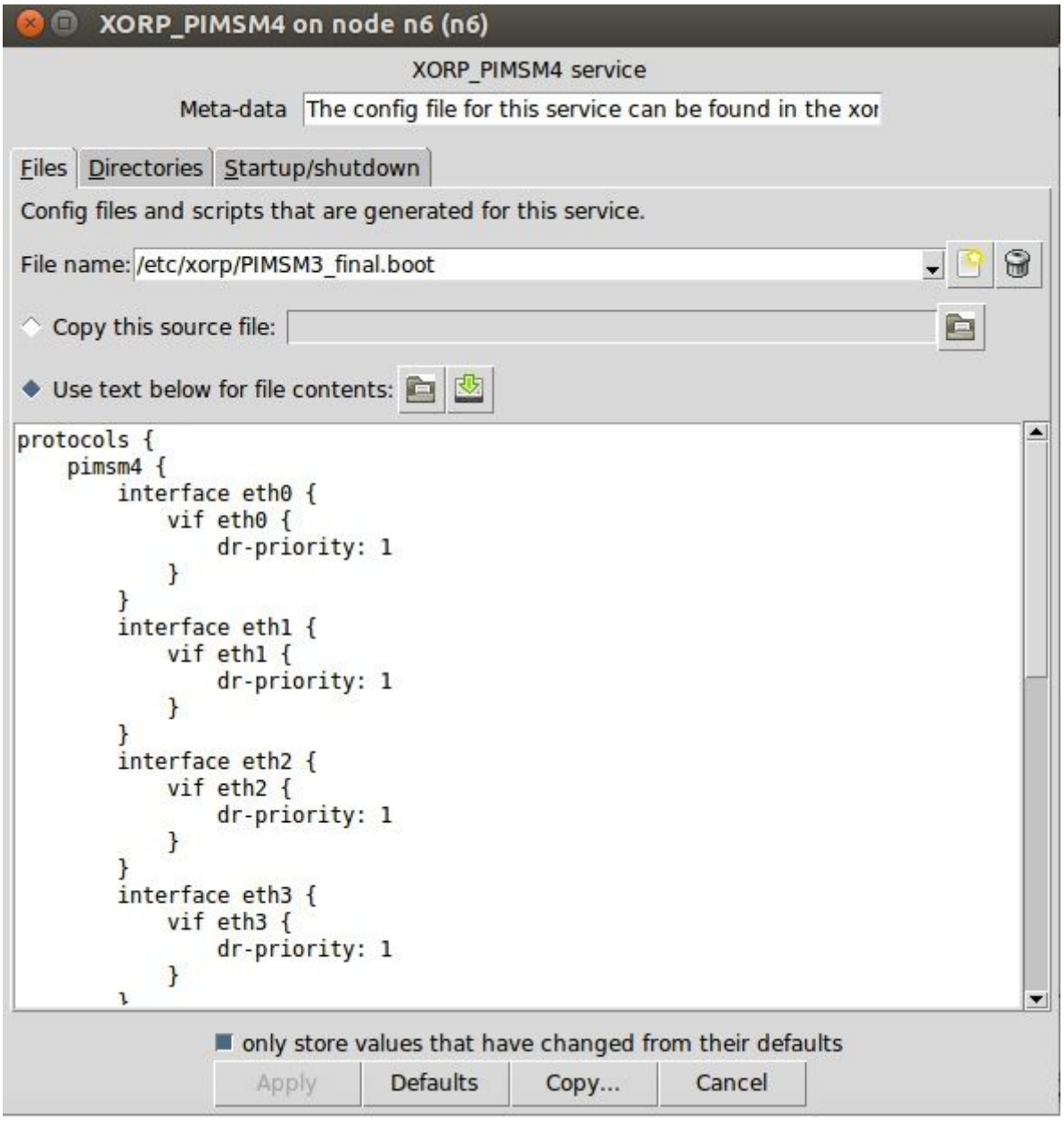

Figura 3.8: Configuración servicio PIM-SM

Se necesita habilitar el servicio xorp_rtrmgr para la configuración del tráfico multicast cuyo archivo de configuración se rellena por defecto y no debe cambiarse. En este

archivo se encuentran directivas para especificar las interfaces y habilitar el protocolo IGMP.

3.1.3. Conectando con la máquina anfitriona

Se puede conectar los nodos virtuales de la herramienta CORE con la máquina anfitriona, y así poder lanzar aplicaciones con interfaz gráfica. Hay varios modos de conectar los nodos con la máquina o viceversa. Un método es usar un nodo RJ45 junto con una interfaz dummy, y el otro es el control de red.

Para conectar un nodo con la máquina anfitriona mediante el nodo RJ45 primero se debe cargar la interfaz dummy. Una vez hecho esto se debe asignar esa interfaz al nodo RJ45 y empezar la sesión de emulación. Finalmente se debe asignar la dirección IP de la máquina host al nodo enlazado con el RJ45. Las líneas de comandos a ejecutar en el terminal de la máquina host son los siguientes:

```
# carga la interfaz dummy
sudo modprobe dummy numdummies=1
# se determina el nombre del puente para la interfaz dummy
sudo brctl show
# se asigna la dirección IP al nodo enlazado
sudo ip addr ad 10.0.2.15/24 dev b.48404.39651
```

El otro método es el modo más rápido de conectar con la máquina anfitriona, que se basa en activar el control de red primario como se muestra en la figura 3.9 . Esto se configura estableciendo un prefijo de control en la ventana que aparece en la opción Options del menú Sessions. Si se define el prefijo de control 192.168.1.0/24 se crea un puente en la máquina host teniendo ésta la última dirección del prefijo, 192.168.1.254, y cada nodo tendrá una interfaz de control configurada con la dirección correspondiente al número de nodo, es decir, 192.168.1.5 para el nodo 5. De esta forma se puede ejecutar una aplicación X11 en el nodo mediante SSH.

```
ssh -X 192.168.1.5 vlc
```

3.2. Arquitectura de red

El diseño de la red implementada en el emulador se basa en la arquitectura que ofrece Cisco en su página web de soluciones de IPTV para operadores de telefonía, mostrada en la figura 3.10. Esta arquitectura es simplificada en el emulador debido a

que CORE no dispone de implementaciones de diferentes elementos de red hardware y software, como el middleware o la gestión de derechos y acceso condicional.

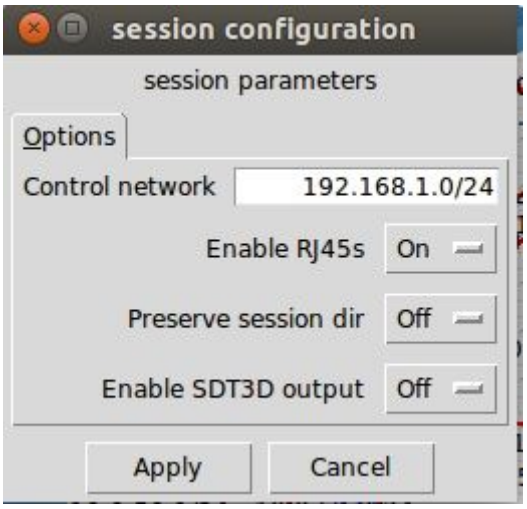

Figura 3.9: Ventana configuración control de red

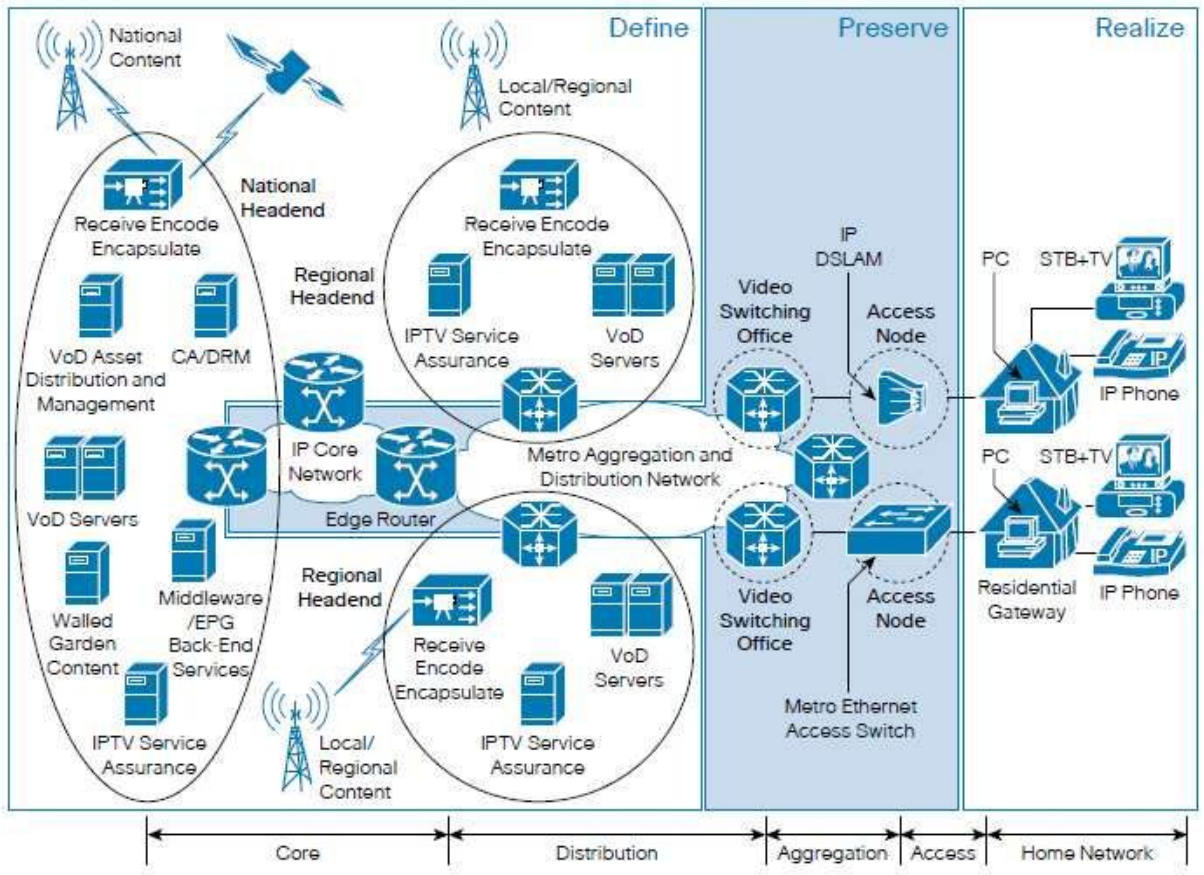

Figura 3.10: Arquitectura de red IPTV [IPTV]

3.2.1. Escenarios de red unicast

Estos escenarios se realizan para experimentar con el emulador probando la configuración de protocolos de enrutamiento, conectando los nodos emulados con la máquina anfitriona y realizando diferentes pruebas detalladas en el capítulo siguiente en cada uno de los escenarios.

Escenario de red conectado a máquina host mediante interfaz dummy

En primer lugar se diseña una red [Figura 3.11] implementada para tráfico unicast mediante Quagga habilitando y configurando el servicio RIP (Routing Information Protocol) en todos los routers. El cliente es un nodo host en el emulador que no requiere configuración, sólo habilitar el servicio SSH. El servidor es la máquina anfitriona que se enlaza con el primer nodo de la red emulada en CORE a través del nodo RJ45 siguiendo los pasos explicados en el apartado 3.1.3. Además de seguir esos pasos se debe añadir una ruta de forma que se sepa que el primer nodo de la red emulada está accesible a través de la máquina host. Esto se hace con la siguiente línea de comando en el terminal de la máquina anfitriona:

```
ip route add 10.0.2.1/24 via 10.0.2.15
```

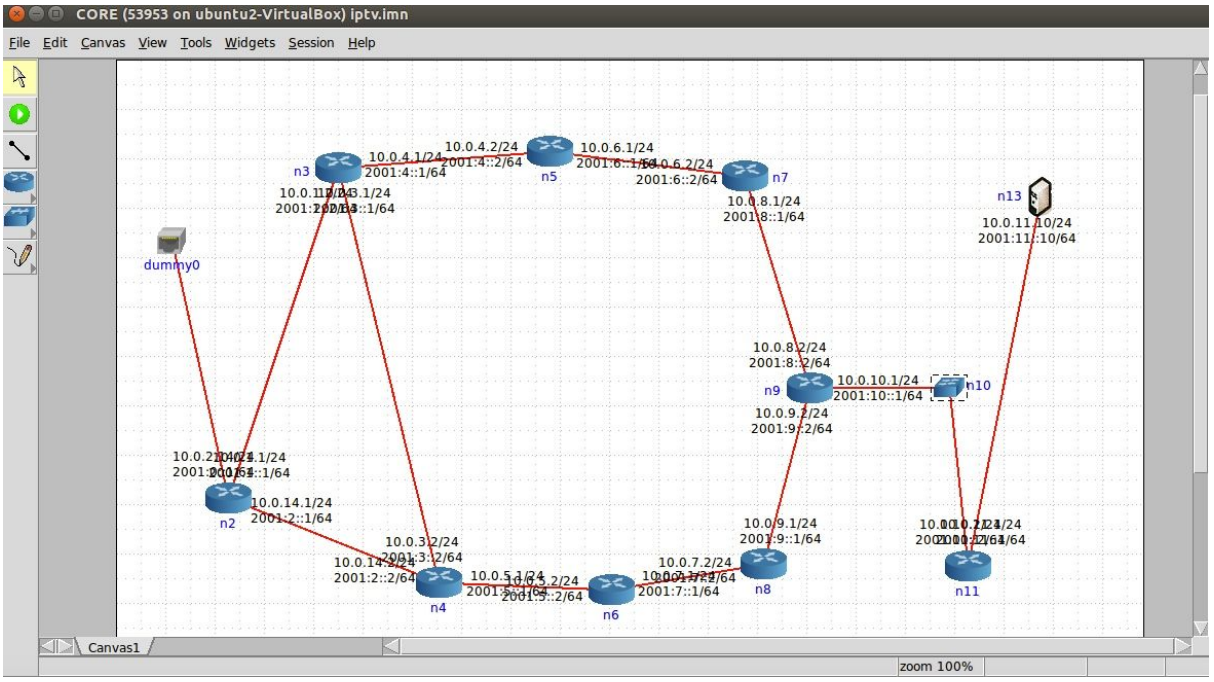

Figura 3.11: Escenario de red con nodo RJ45

Escenario de red con nodo host como servidor

El siguiente escenario de red [Figura 3.12] que se emula para el tráfico unicast difiere con el anterior en que se trata de un servidor implementado a través de un nodo host en el emulador CORE. A partir de este punto se utiliza la activación del control de red primario para conectar la máquina host con el emulador CORE, y así poder lanzar el reproductor VLC en la configuración del servidor y cliente. La dirección introducida para el control de red es 192.168.1.0/24. Aquí se trata de probar el otro método para conectar el nodo emulado con la máquina anfitriona. Se utiliza la configuración del protocolo de encaminamiento RIP.

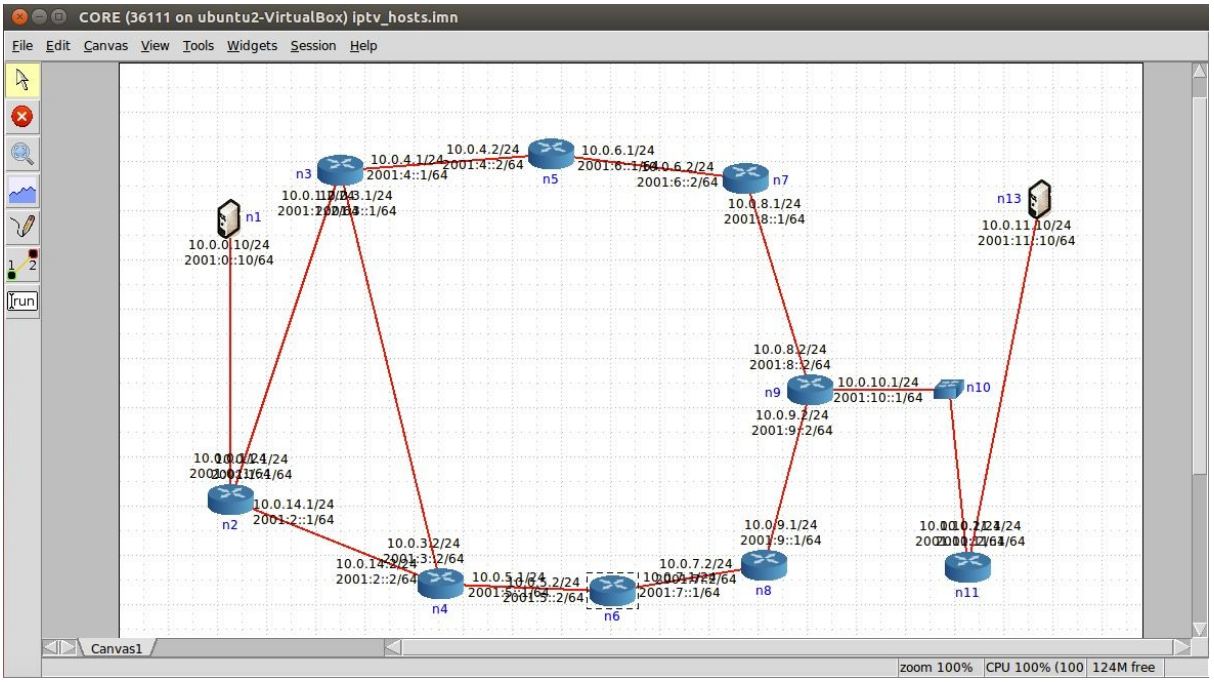

Figura 3.12: Escenario de red con dos hosts como servidor y cliente

Escenario de red con varios hosts clientes

Este es el último escenario de red que se implementa para probar tráfico unicast, y es el presentado en la figura 3.13, en el que se prueba la entrega de tráfico a varios clientes. Este escenario está compuesto por un host que tiene el rol de servidor, una red troncal con topología anillo y una red de acceso conectada a varios hosts clientes. Ambas redes están separadas por un switch, que reenvía los paquetes entrantes a los nodos enlazados utilizando una tabla hash de direcciones Ethernet. No es necesario configurar ningún servicio en el switch.

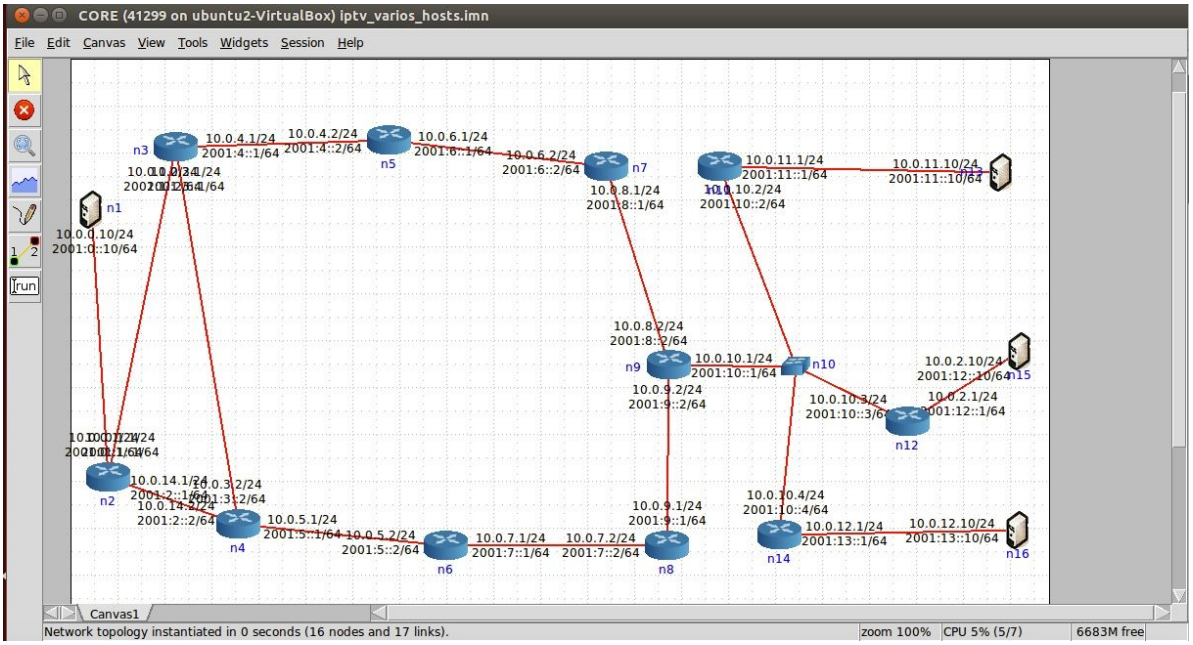

Figura 3.13: Escenario de red con varios hosts clientes

3.2.2 Escenarios de red multicast

Escenario de prueba

Antes de configurar el escenario de red final para tráfico multicast se realiza una prueba de una red sencilla con un host servidor y dos hosts clientes interconectados a través de tres routers para poder comprobar el funcionamiento de los protocolos de enrutamiento y diagnosticar problemas más fácilmente en el caso de que los hubiera. Este escenario se muestra en la figura 3.14. La configuración del tráfico multicast se realiza mediante XORP habilitando los servicios OSPFv2 y PIM-SM4, y realizando los archivos de configuración para cada router. En los nodos host no es necesario realizar ninguna configuración, solo debe estar habilitado el servicio SSH. Se usa la red de control para tener conectividad con la máquina anfitriona.

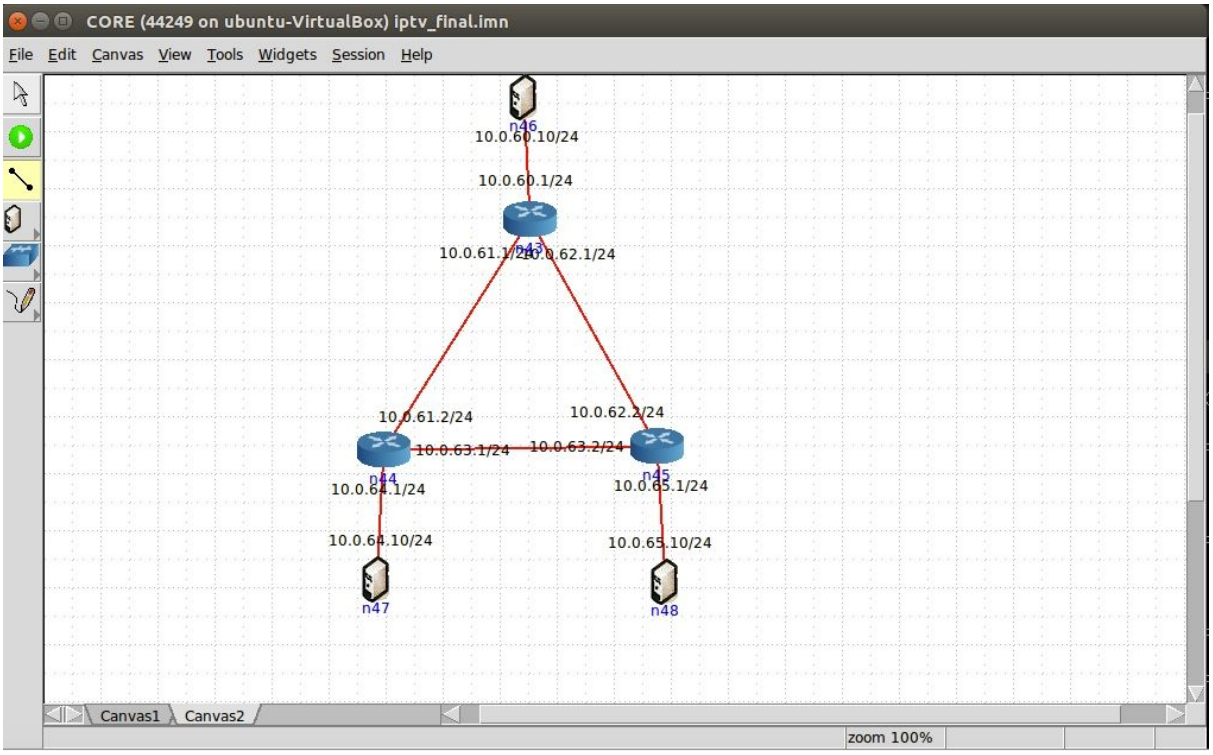

Figura 3.14: Escenario de prueba multicast

Escenario final

El escenario de red final [Figura 3.15] configurado para el tráfico multicast está formado por una red troncal que se trata de la topología Noruega formada por 27 nodos. Se incluyen más hosts clientes cuya red de acceso está conectada a otro router de la red troncal, de forma que se tenga que dividir el tráfico multicast. En este escenario también obtenemos la conectividad con la máquina anfitriona mediante la activación del control de red.

En la configuración de los routers directamente conectados con los hosts clientes se deshabilitan las interfaces correspondientes puesto que los host no intercambian paquetes OSPF o PIM y se configura una policy para que esas interfaces sean anunciadas a los protocolos de enrutamiento. Esto no funciona como se espera puesto que al realizar capturas de tráfico en el host se observa que le llegan estos paquetes. Esto puede deberse a la configuración del servicio xorp_rtrmgr que incluye ambos protocolos y no se puede modificar.

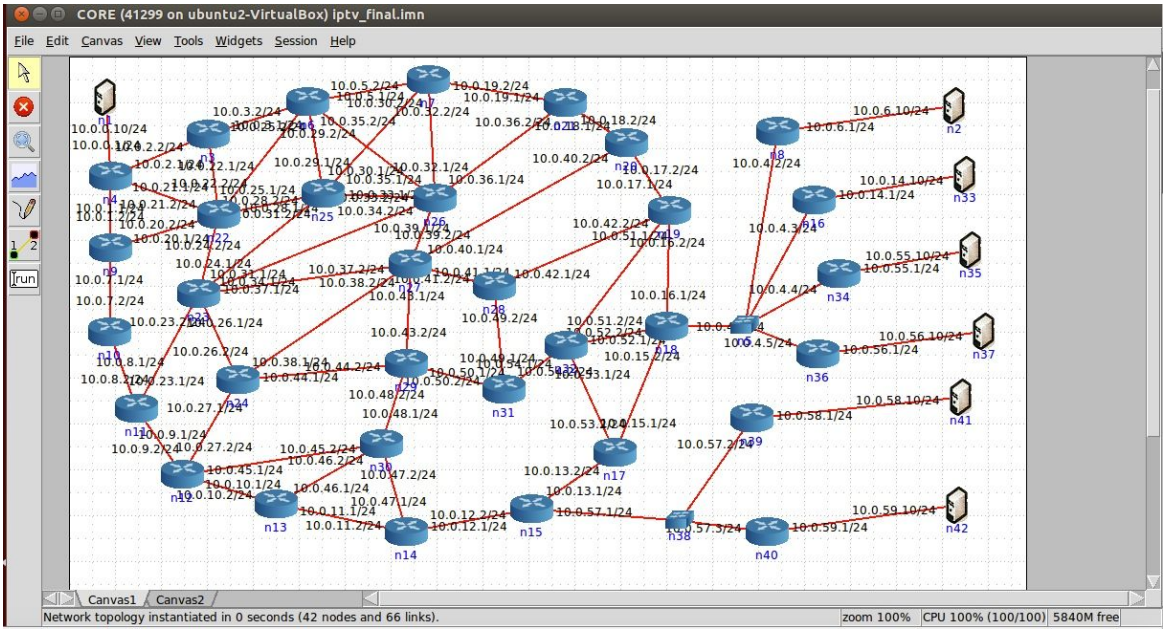

Figura 3.15: Escenario final multicast

3.3.- Aplicación de distribución de vídeo

La aplicación de distribución de vídeo está basada en el reproductor y framework VLC media player, que permite enviar vídeo y comprobar su entrega en otra ventana del reproductor. De forma que conectando la máquina anfitriona con los nodos hosts que hacen de servidor y clientes se puede lanzar el reproductor VLC mediante SSH. Se decide conectar la máquina con los hosts mediante la activación del control de red primario. Para ejecutar VLC en los nodos de CORE se ejecutan los siguientes comandos en el terminal de la máquina anfitriona especificando la dirección IP del nodo emulado, donde la opción -X redirige la salida gráfica a la máquina anfitriona:

```
ssh -X 192.168.1.1 vlc
```

Una vez lanzado VLC en cada host se procede a configurar el reproductor de forma que el servidor emite un vídeo y los clientes lo reciben. La configuración de VLC para emitir y recibir el flujo de vídeo es detallada en los anexos.

Capítulo 4

RESULTADOS Y EVALUACIÓN

En este capítulo se presentan las diferentes pruebas que se han realizado en los escenarios emulados que se basan en comprobar la conectividad de la red y la entrega de vídeo mediante la configuración del framework VLC media player. También se detallan dos casos de estudio para probar la infraestructura final desarrollada.

4.1. Entrega tráfico unicast

4.1.1. Escenario de red conectado a máquina anfitriona mediante interfaz dummy

Este fue el primer escenario definido en el capítulo anterior (véase figura 3.11). La primera prueba realizada se basa en comprobar la conectividad entre la

máquina host y la red emulada. Para ello se realiza un ping con éxito a la dirección IP de la interfaz que conecta el nodo RJ45 con el primer nodo de la red.

ping 10.0.2.1

Por otra parte para comprobar el correcto funcionamiento del protocolo de enrutamiento unicast se realiza un ping a diferentes nodos del escenario de red emulado dentro del CORE, entre ellos el host cliente.

Finalmente se prueba la entrega de un flujo de vídeo, para lo cual se abre el reproductor VLC en la máquina anfitriona, y se configura para emitir un archivo de vídeo a la dirección IP del host cliente, en el que se ejecuta VLC mediante SSH como se muestra en la figura 4.1, y se configura para recibir el flujo de vídeo. Se concluye que la recepción y calidad de la imagen es aceptable con algunas imágenes con efecto bloque o pixeladas pero con una reproducción continua. El resultado de este proceso es el mostrado en la figura 4.2 y figura 4.3.

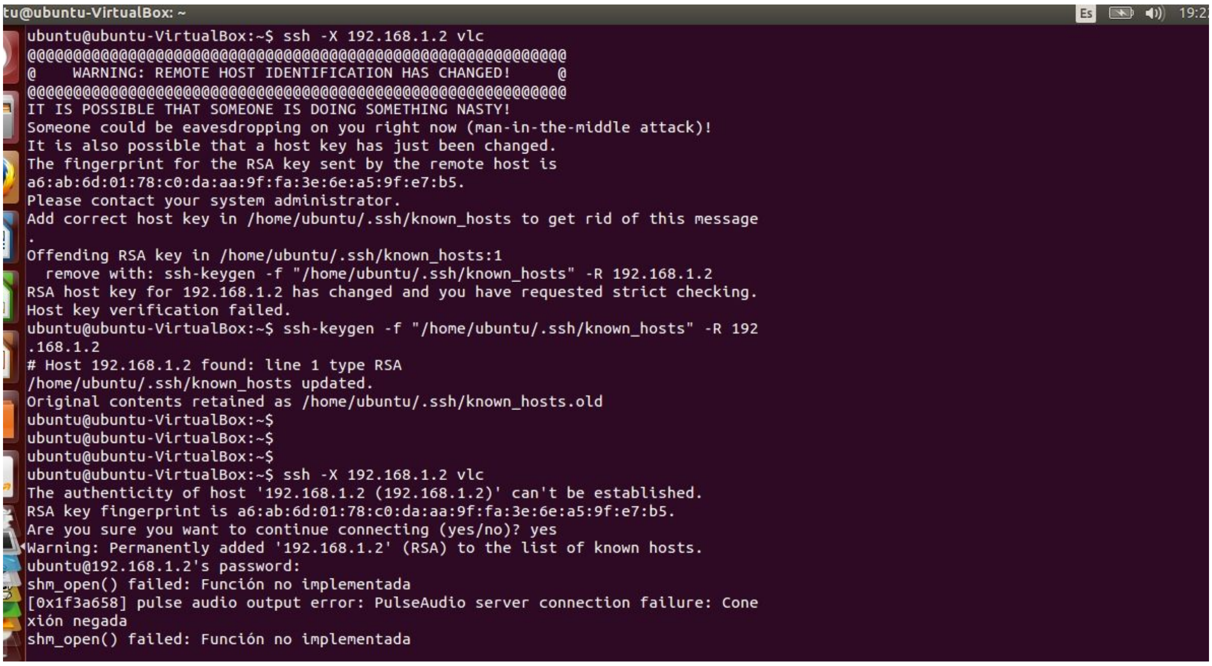

Figura 4.1: Lanzamiento VLC en el host cliente

Figura 4.2: Servidor emitiendo flujo de vídeo

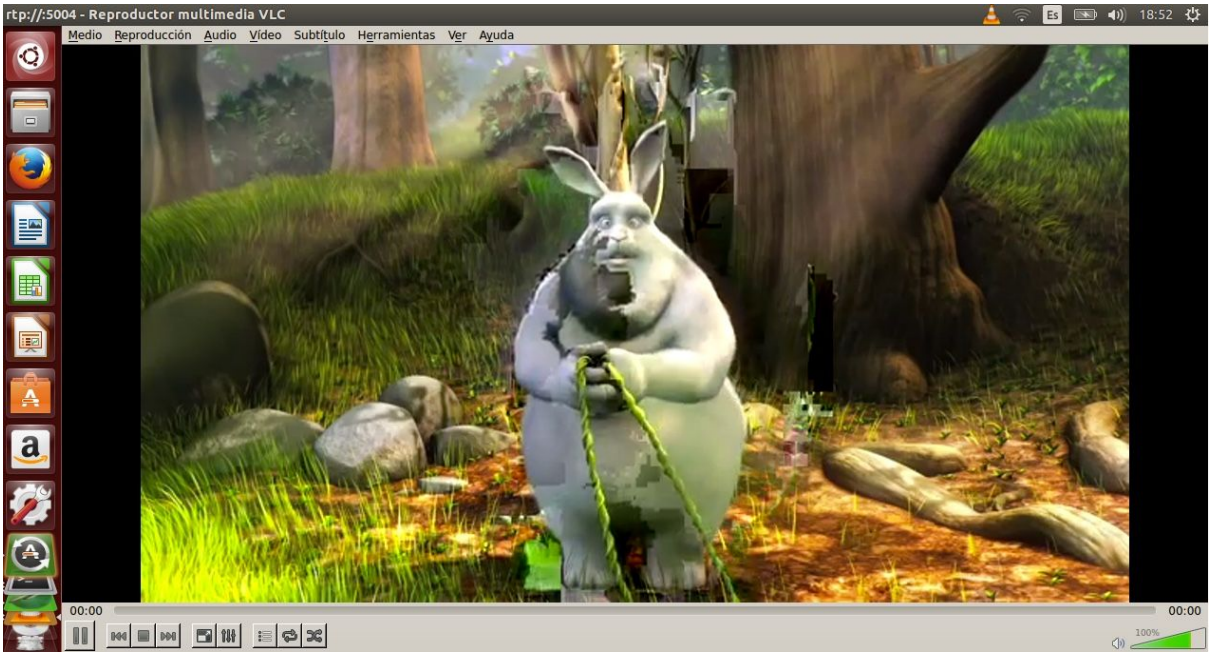

Figura 4.3: Cliente recibiendo flujo de vídeo

4.1.2. Escenario de red con nodo host para servidor

Aquí se prueba el segundo escenario presentado en apartado 3.2.1 del capítulo anterior para tráfico unicast (véase figura 3.12). La primera prueba que se realiza en este escenario es la realización de varios ping para comprobar la conectividad en el escenario de red, entre varios nodos de la red y de nodo host servidor a nodo host cliente; todos ellos con éxito.

Finalmente se prueba el envío de un flujo de vídeo a la dirección IP del host cliente, ejecutando el reproductor VLC, que se lanza en ambos nodos host mediante SSH como se muestra en las figuras 4.4 y 4.5, y configurando la emisión y la recepción. Se observa una buena calidad de imagen y una reproducción continua. El resultado se muestra en la figura 4.6 y 4.7.

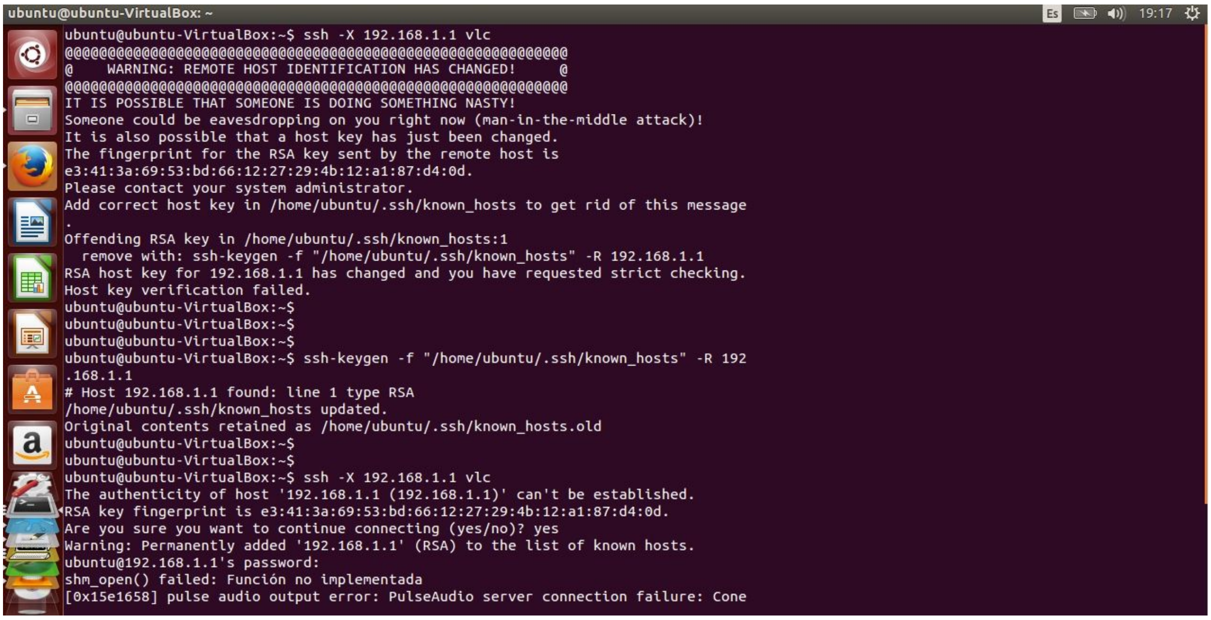

Figura 4.4: Lanzamiento VLC en el host servidor

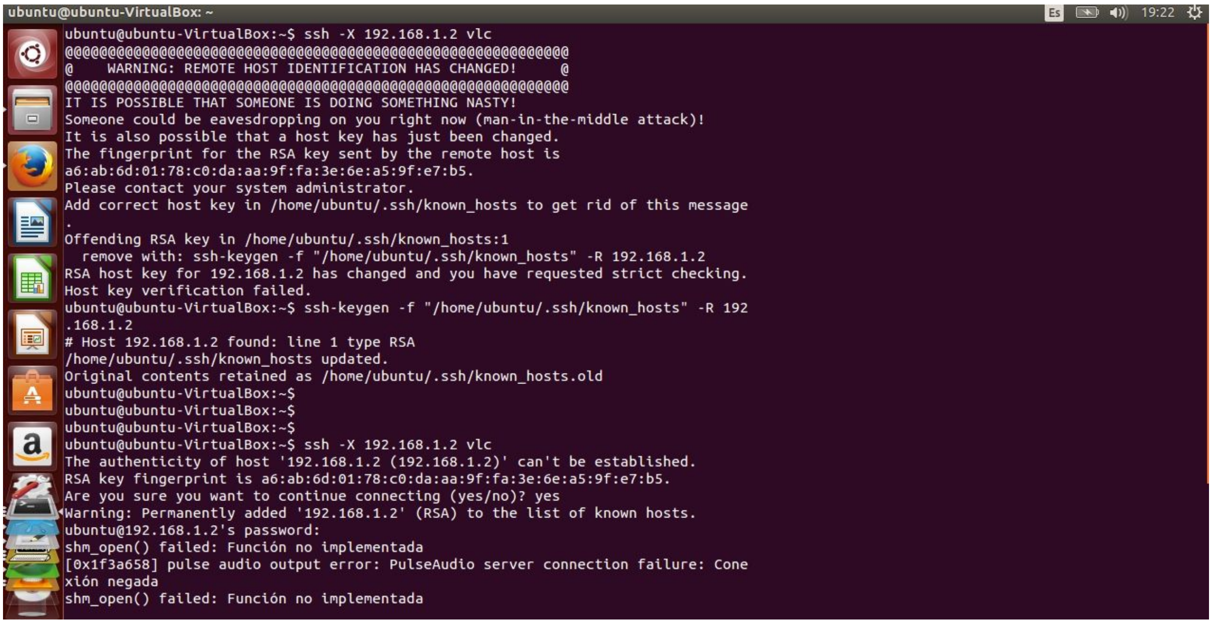

Figura 4.5:Lanzamiento VLC en el host cliente

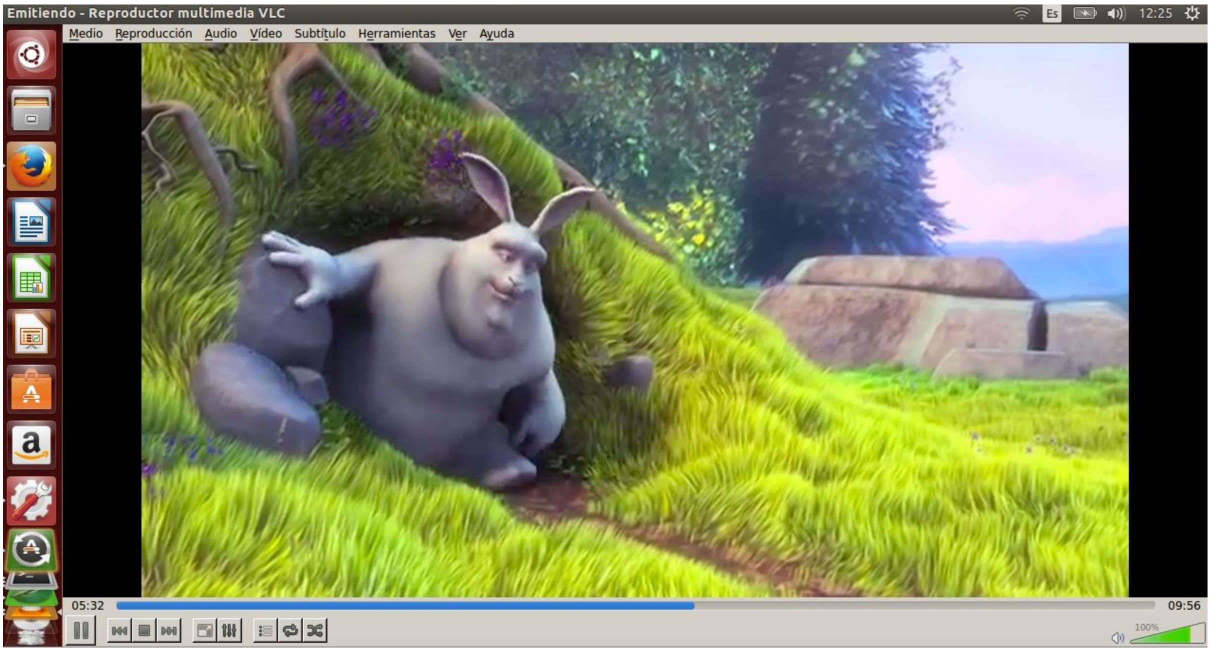

Figura 4.6: Servidor emitiendo flujo de vídeo

Figura 4.7: Cliente recibiendo flujo de vídeo

4.1.3. Escenario de red con varios hosts clientes

Este es el último escenario de red que se configura para probar tráfico unicast, escenario presentado en el capítulo anterior (véase figura 3.13). Se vuelven a realizar las mismas pruebas que anteriormente, comprobando la conectividad mediante un ping a cada host cliente antes de pasar a la entrega de vídeo mediante el reproductor VLC. Es necesario ejecutar tres lanzamientos del reproductor VLC en el nodo host servidor para configurar el envío del flujo de vídeo a cada dirección IP correspondiente a cada host cliente. El ping a cada host es realizado con éxito y el flujo de vídeo recibido en cada cliente es, en general, de buena calidad con algún efecto bloque y congelamiento de la imagen en algún momento de la reproducción.

4.2. Entrega tráfico multicast

4.2.1. Escenario de prueba

En este apartado se realizan pruebas al primer escenario definido en el apartado 3.2.2. (véase figura 3.14). Una vez configurada esta red se comprueba el correcto funcionamiento de los protocolos de enrutamiento. Para ello se ejecuta el comando ping de host servidor a ambas hosts clientes, y se realiza un prueba

mediante iperf, herramienta que se utiliza para medir el rendimiento de la red, de forma que podemos comprobar que se realiza una entrega de paquetes a una dirección multicast entre los nodos hosts servidor y clientes. Esta prueba se realiza iniciando primero el servidor, que es el que recibe los datos, y después el cliente, especificando en ambas líneas de comando la dirección multicast y que se envían datagramas UDP. En el lado del cliente también se debe especificar el valor del TTL mediante la opción -T. Estas pruebas se hacen en los nodos host n1, n2 y n3, que hacen de servidor y clientes respectivamente; las salidas de comandos se muestran en las figuras 4.8 y 4.9 Las líneas a ejecutar son las siguientes:

```
iperf -s -u -B 225.0.1.2 -i 1
iperf -u -c 225.0.1.2 -T 4
```

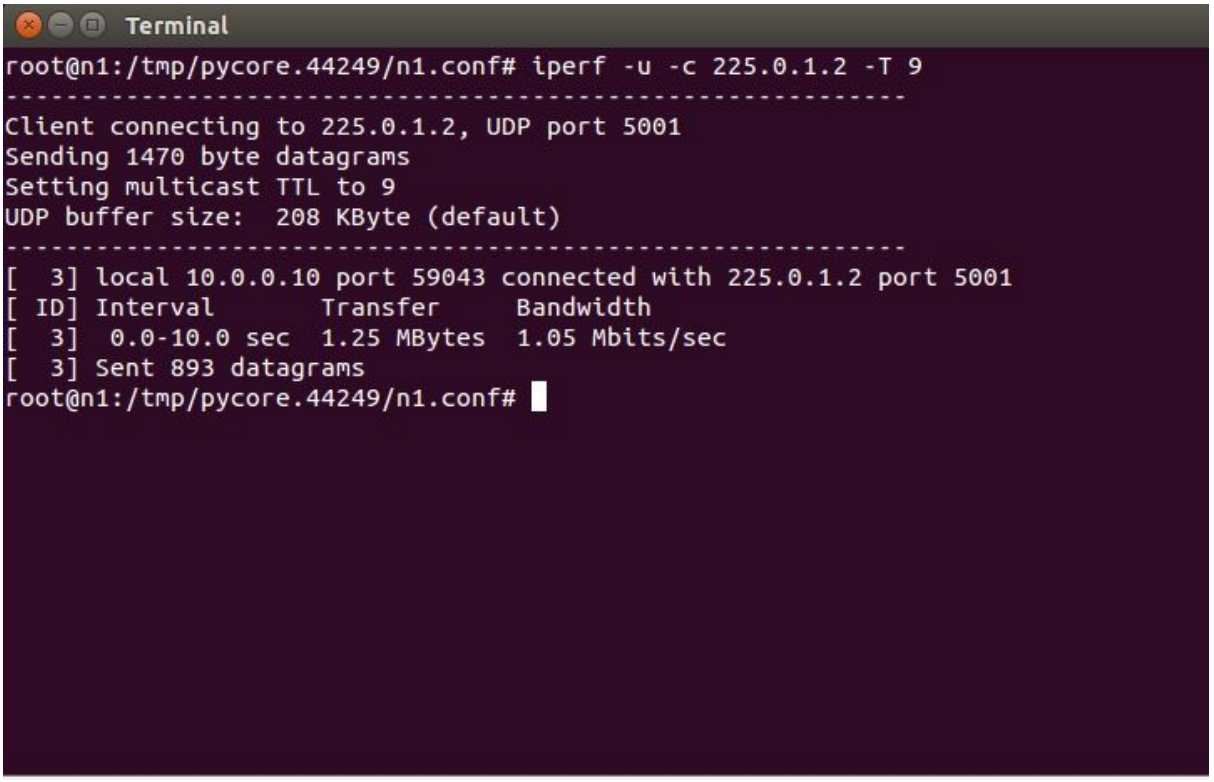

Figura 4.8: Comando iperf como cliente en el host servidor n1

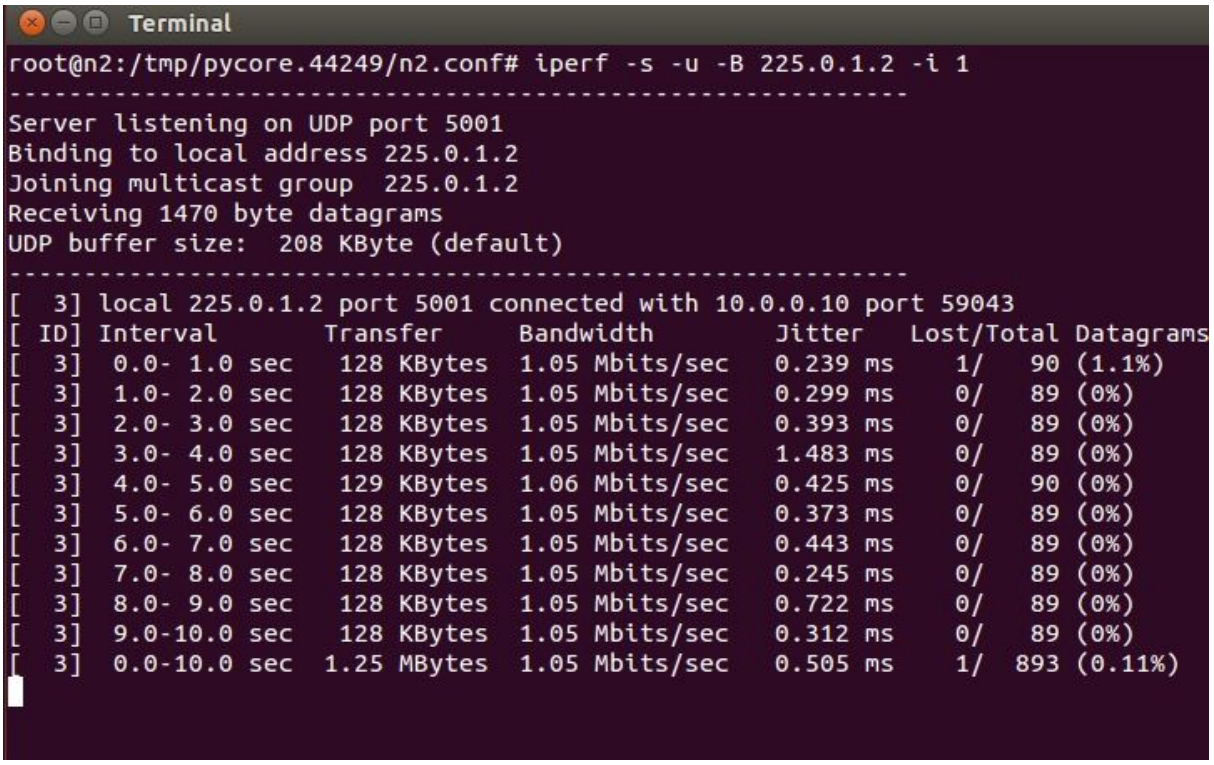

Figura 4.9: Comando iperf como servidor en el nodo host n2

4.2.2. Escenario final multicast

El escenario final para tráfico multicast es el presentado en el apartado 3.2.2 (véase figura 3.15). Acabada la configuración se procede a realizar las mismas pruebas anteriores entre el nodo host servidor y los seis nodos hosts clientes, y por último, la emisión de un flujo de vídeo a través de la aplicación de distribución de vídeo basada en el reproductor VLC.

En este escenario de red se comprueba una peor recepción del flujo de vídeo observando la imagen pixelada y/o congelada; la reproducción no es fluida. Se advierte una diferencia, si se añade en la configuración de la emisión, el proceso de transcodificación en la que se observa una recepción más congelada y peor calidad de la imagen. Estos problemas pueden deberse a la carga computacional que supone todas las recepciones de vídeo simultáneas. También hay que tener en cuenta que el rendimiento puede verse afectado si se ejecuta CORE en una máquina virtual, puesto que los recursos se pueden reducir al tener que realizar también la virtualización de los nodos emulados. En la figura 4.10 se muestra un momento de la recepción del flujo de vídeo en el cliente, en el que se paró la reproducción hasta unos segundos más tarde.

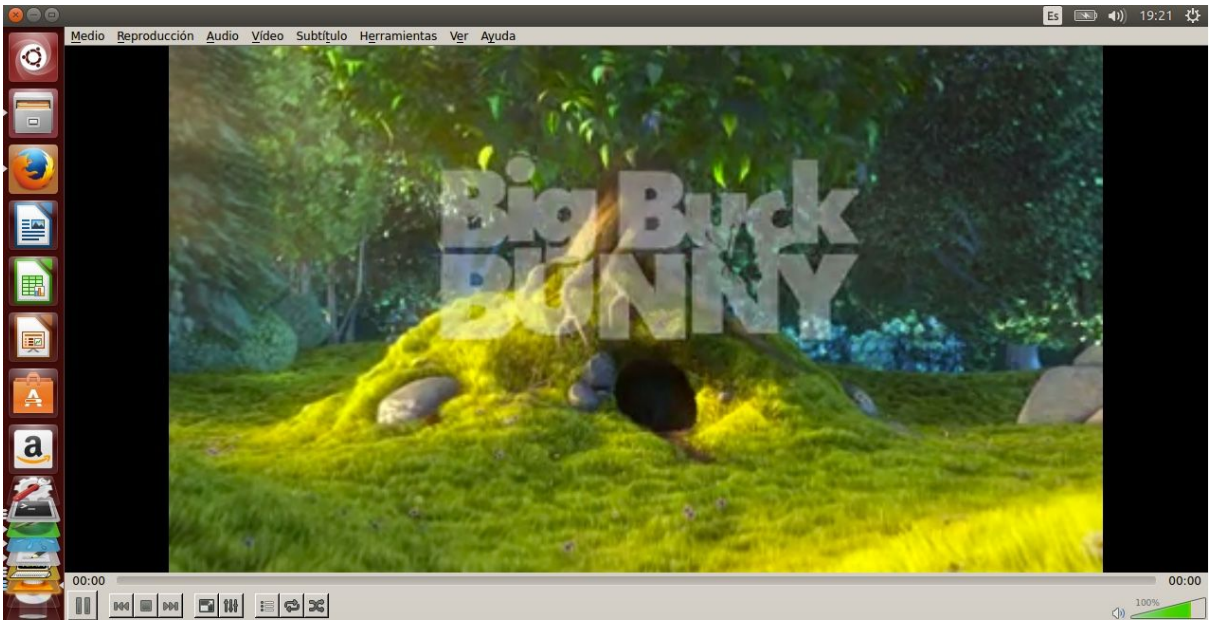

Figura 4.10: Cliente recibiendo flujo de vídeo

4.3. Casos de estudio en entorno de pruebas

Se realizan dos casos de estudio para demostrar el tipo de experimentos que se pueden hacer con la plataforma desarrollada.

4.3.1. De tráfico multicast a unicast[3]

Los proveedores de servicios ofrecen la posibilidad de pausar la reproducción del contenido o volver al inicio de programa, por lo que se enfrentan al problema de dejar de enviar los paquetes a través de multicast para enviar el contenido personalizado mediante transmisión unicast al usuario interesado. De manera que es determinante el dimensionado de la red para evitar que se degrade la calidad de la imagen y con ella la calidad de experiencia (QoE).

Se ha reproducido esta situación en el emulador mediante la herramienta iperf en un escenario con cuatros nodos hosts clientes, de forma que se transmite un flujo de paquetes simulando un codificador de vídeo que genera vídeo de alta definición a 15 Mbps, primero a través de multicast para luego pasar a tráfico unicast. Para ello,

[3] Los dos casos de estudio se realizan, por fallos técnicos, con un escenario de red anterior al aquí presentado, en el que no se produce separación de tráfico con dos hosts clientes más conectados a otro router de la red troncal.

después de iniciar la sesión de emulación, se inicia el servidor escuchando en una dirección multicast en cada host cliente. Después se configura el cliente a la misma dirección multicast indicando con la opción -b la cantidad de tráfico que se envía, y con la opción -t se indica el tiempo de duración de la medida. También se debe configurar servidor y cliente para tráfico unicast como paso previo a la transmisión, por lo que se configura en cada host cliente un servidor iperf, en la línea de comando no es necesario indicar la dirección IP unicast; y en el nodo host servidor se escribe una línea de comandos para configurar la transmisión para cada host cliente indicando su dirección IP correspondiente. Se deben abrir tantas terminales en el host servidor como clientes para configurar el tráfico unicast. La transmisión en ambos casos es sobre el protocolo UDP. En las figuras 4.11 y 4.12 se muestran las configuraciones del lado del cliente en el host servidor, el que envía los paquetes, para el tráfico unicast y multicast.

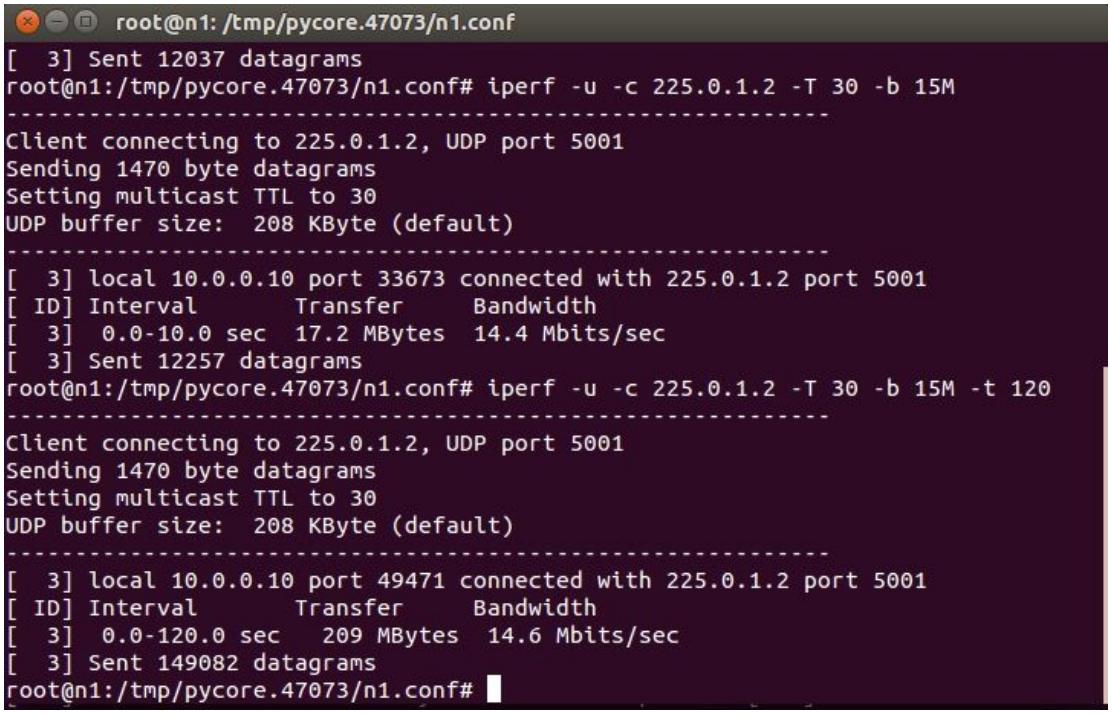

Figura 4.11: Comando iperf cliente en el host servidor para multicast

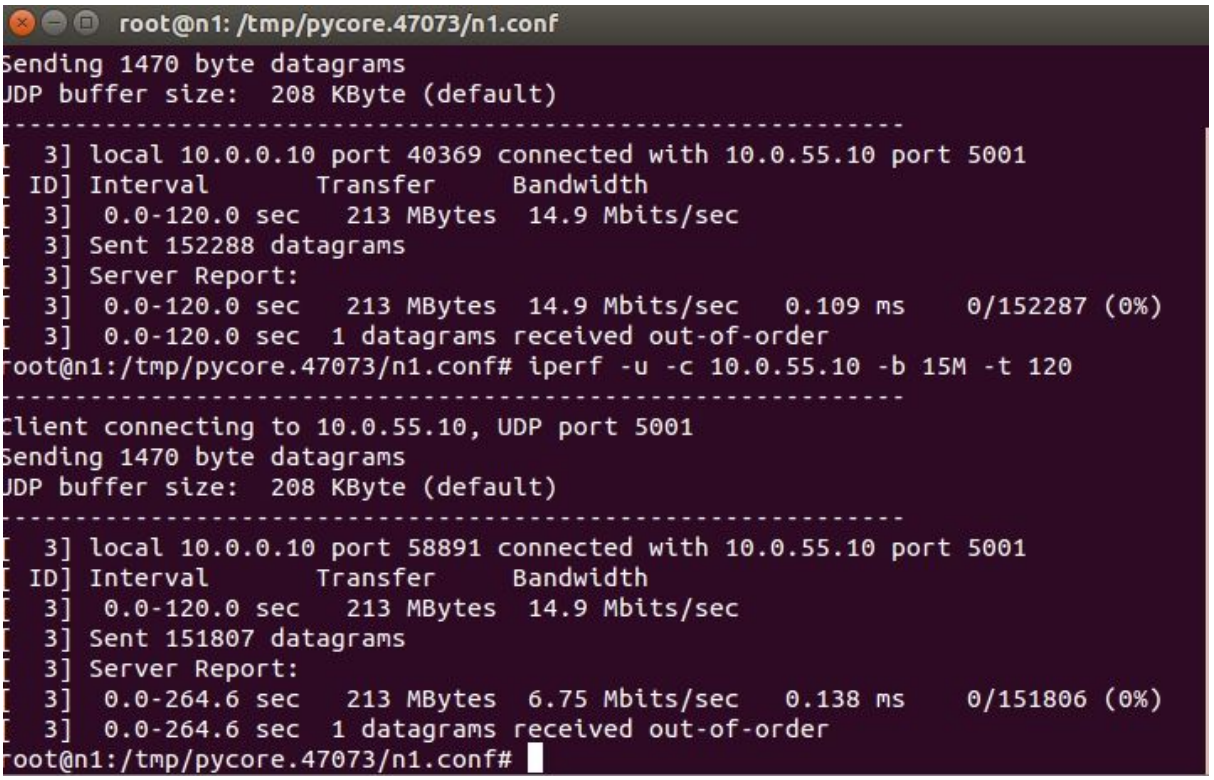

Figura 4.12: Comando iperf cliente en el host servidor para unicast

Una vez preparadas las líneas de comando en cada terminal se procede a ejecutarlas empezando con la transmisión a la dirección multicast elegida. Pasado cierto tiempo se comienza la ejecución de cada cliente iperf configurado en el host servidor a cada dirección IP unicast paulatinamente. En las terminales de cada host cliente se muestran los resultados de la medida cada segundo; se observa que aproximadamente el primer medio minuto de transmisión a la dirección unicast no se recibe ningún paquete; después los datos son recibidos normalmente. Del resumen [Figura 4.13] mostrado por el cliente iperf tras finalizar la transmisión se observa que ha habido un 0% de paquetes perdidos, porcentaje en el que no se incluyen aquellos que no han llegado. Por otra parte, los resultados obtenidos en el tráfico multicast muestran una pérdida de paquetes de 0.042%, que se puede considerar buena, ya que para mantener una buena calidad del enlace la pérdida de paquetes no debe ser mayor del 1%. En las figuras 4.14 y 4.15 se muestran los resultados obtenidos en un host cliente.

Se vuelve a repetir el proceso varias veces iniciando otra sesión de emulación para ver cómo pueden variar los resultados. En todos los casos, el resultado para el tráfico unicast es el mismo, no llegan datagramas durante la primera parte de la transmisión, pero en la transmisión multicast se observa un porcentaje de pérdida

de paquetes variable, obteniendo como valores 0,5%, 20%, 33% y 90%. No se consigue determinar por qué esta diferencia.

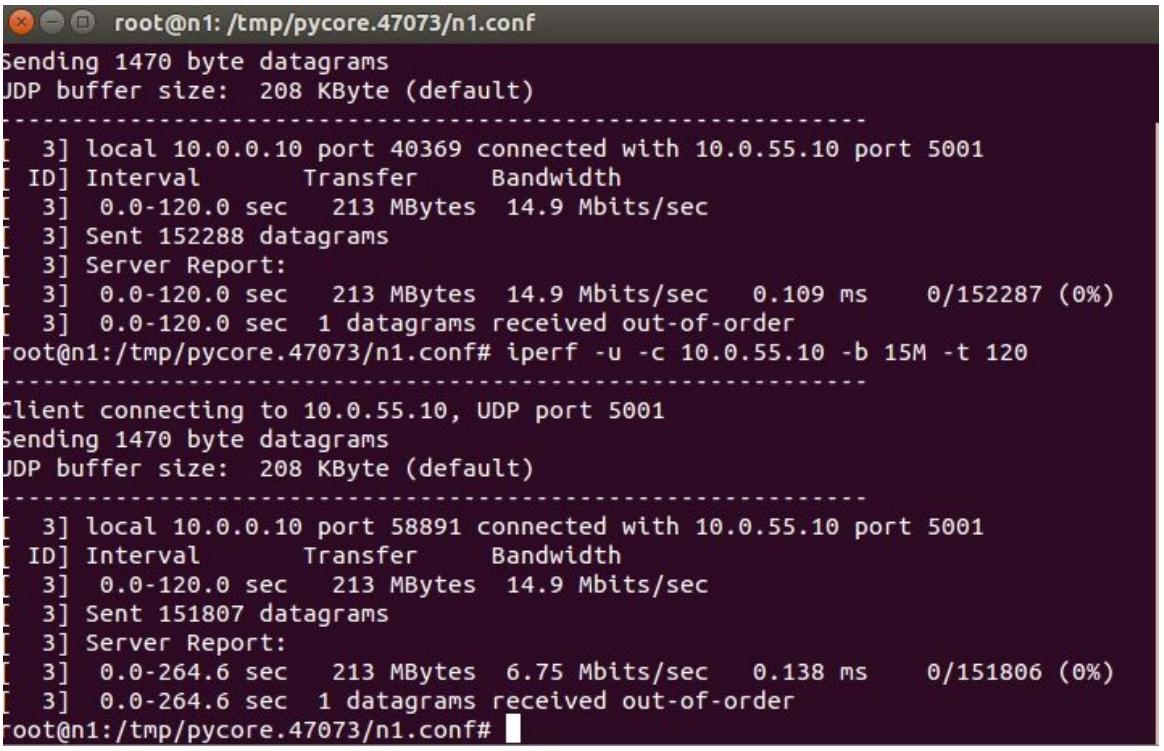

Figura 4.13: Salida de comandos del cliente iperf en el nodo host servidor de unicast

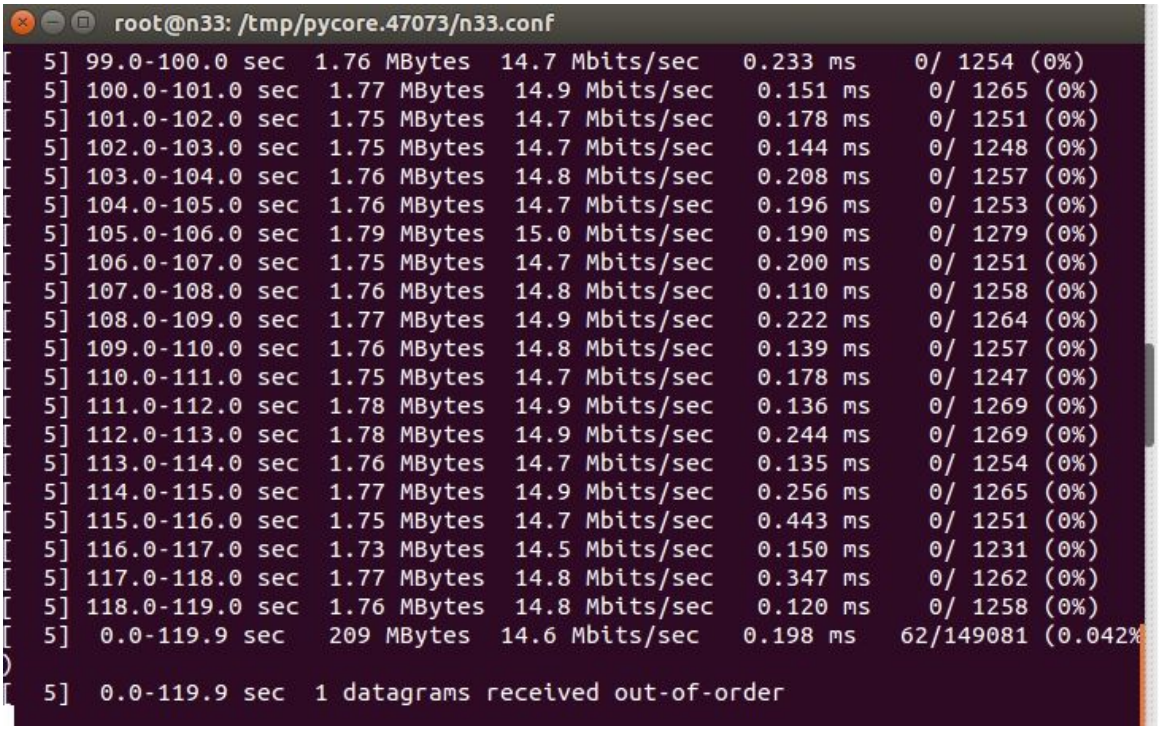

Figura 4.14: Salida de comandos del servidor iperf en el host cliente de multicast

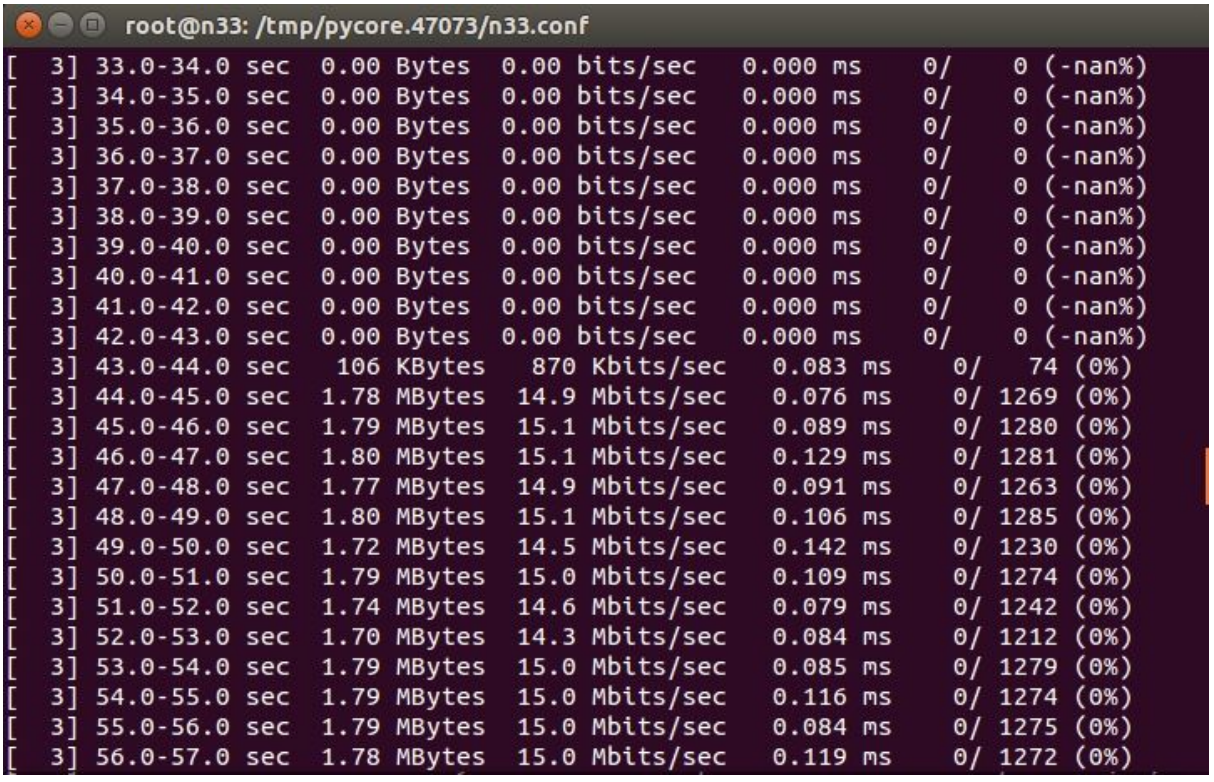

Figura 4.15: Salida de comandos del servidor iperf en el nodo host cliente de unicast

En la figura 4.16 se muestra el camino que siguen los paquetes durante la transmisión marcado con trazo ancho y las gráficas del tráfico que pasa por los enlaces elegidos con la herramienta Plot.

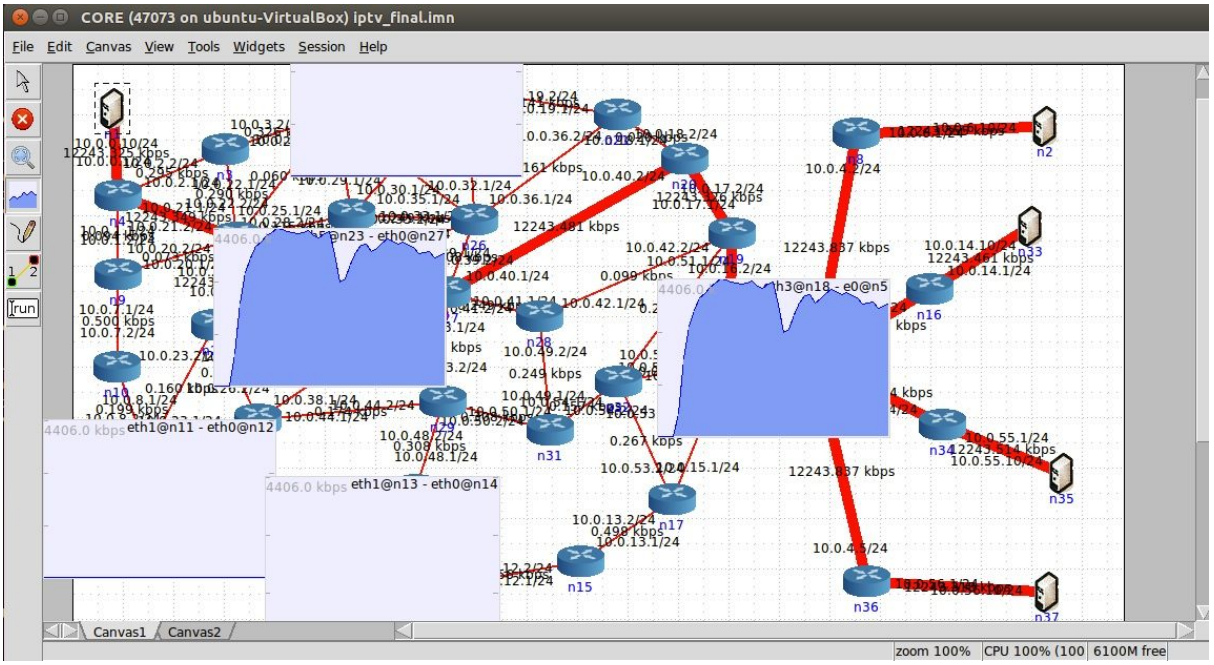

Figura 4.16: Escenario de red CORE durante prueba iperf

Como se puede observar en la última figura los paquetes son replicados en el nodo router correspondiente para la entrega a cada nodo host cliente. Este caso de estudio presentado no es riguroso, se trata de un tipo de análisis que la plataforma desarrollada permite hacer.

4.3.2. Plano de datos

En este caso de estudio se muestra el intercambio de paquetes capturados con Wireshark entre el host cliente y el router directamente conectado, es decir, el router designado (DR), durante la transmisión de un flujo de vídeo. También se estudia que todo sigue funcionando correctamente si se cambia el mecanismo de determinar el router de punto de encuentro (RP) de automático a estático. Para ello se deben cambiar los archivos de configuración de cada router eliminando las líneas de código referentes al mecanismo bootstrap y sustituyéndolas por las líneas que configuran el router RP de manera estática. En la figura 4.17 se muestra la configuración del RP estático.

Lo primero es lanzar el reproductor VLC en el nodo host servidor y nodo host cliente mediante SSH, y abrir Wireshark en cada interfaz. Después se procede a configurar la emisión del flujo de vídeo en el nodo host servidor y la recepción en el nodo host cliente. Los paquetes RTP llegan sobre UDP, por lo que hay que indicar en Wireshark que los decodifique como tal eligiendo los puertos de origen y destino, y el protocolo.

Este procedimiento se realiza en primer lugar con la configuración inicial en la que el router RP se determina mediante el mecanismo bootstrap. En esta configuración el router designado recibe el mensaje IGMP Leave Group del host cliente, para abandonar el grupo multicast al que estaba previamente unido por pruebas anteriores. De manera que el router DR envía el mensaje IGMP Membership Query con destino al grupo multicast que se quiere abandonar. Después recibe el mensaje IGMP Membership Report procedente del host cliente para unirse al nuevo grupo multicast. Después de esto se empiezan a recibir los paquetes con el flujo de vídeo correspondiente, codificados como MPEG TS (Transport Stream). En las figuras 4.18 y 4.19 se muestra el intercambio de paquetes capturados con WIreshark.

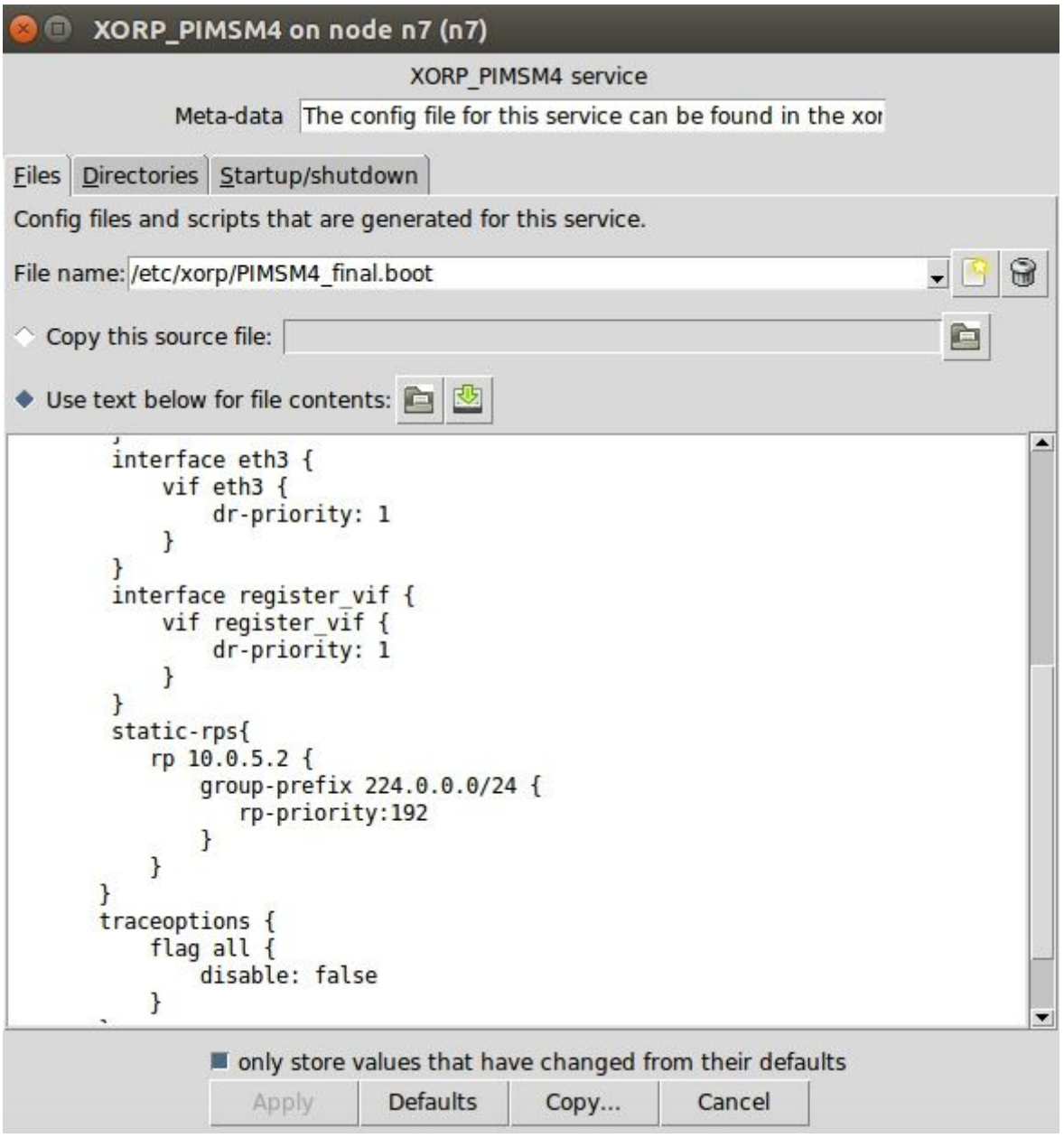

Figura 4.17: Configuración de RP estático

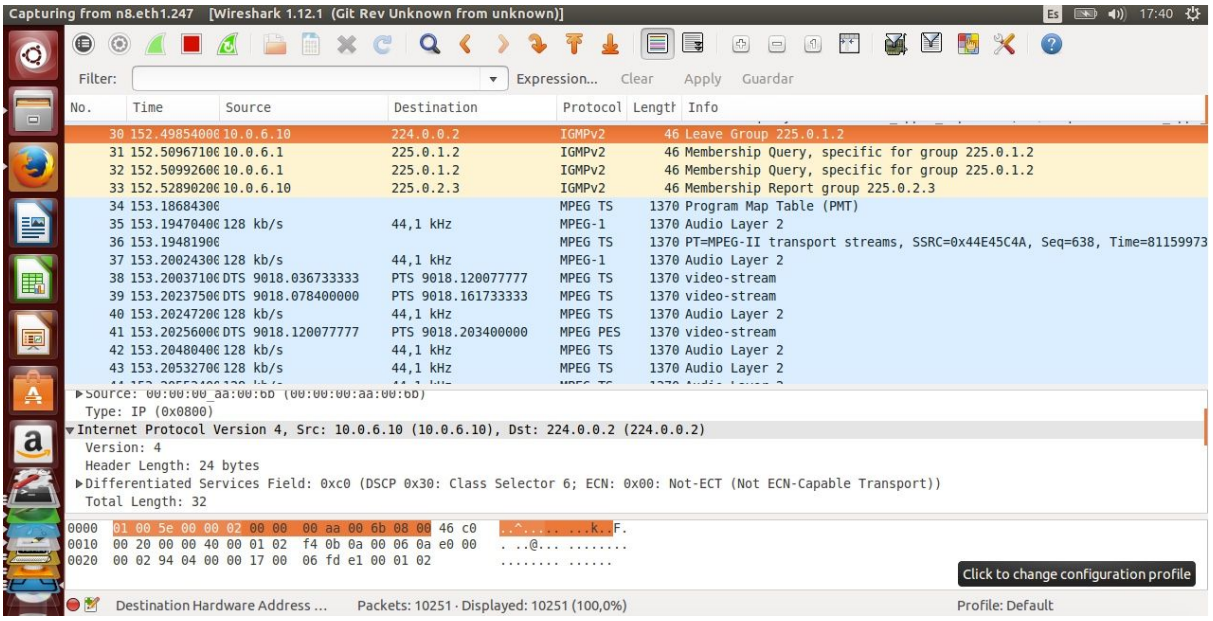

Figura 4.18: Primeros paquetes capturados en el router DR

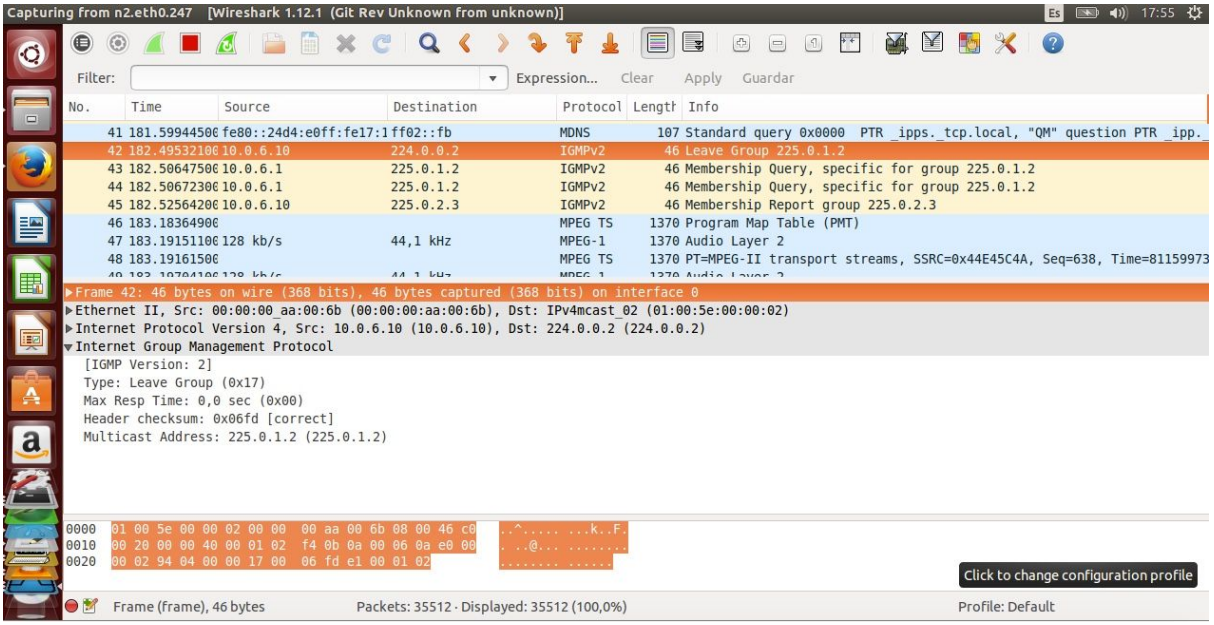

Figura 4.19: Primeros paquetes capturados en el nodo host cliente

Lo siguiente es probar que sigue funcionando correctamente si se determina de manera estática el router punto de encuentro, RP. Por lo que se procede a cambiar los archivos de configuración del servicio PIMSM4, donde se configura con una prioridad mayor de convertirse en router RP a los primeros routers del interior de la red troncal. Antes de pasar a transmitir un flujo de vídeo y capturar los paquetes, se realiza una prueba iperf en la que se observa que durante los primeros instantes no

llega nada puesto que el árbol de distribución está en proceso, y se observan diferentes trazos marcados hasta que se observa el mismo camino que en el mecanismo bootstrap por el que fluyen los paquetes, como se puede ver en la figura 4.20.

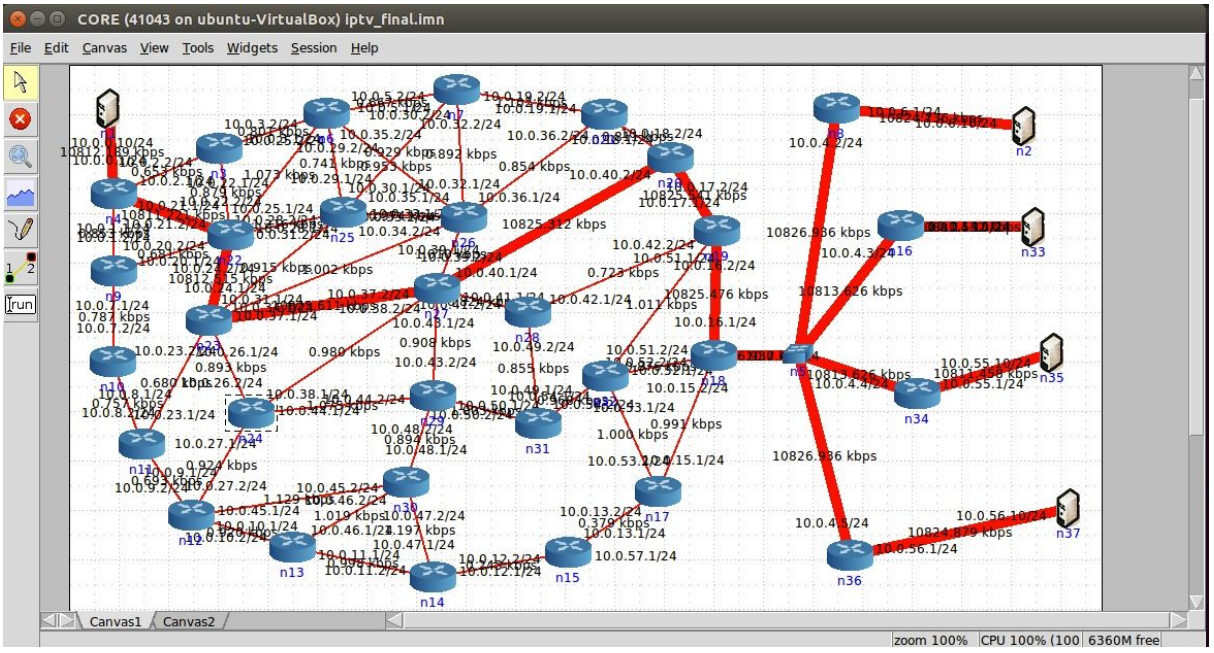

Figura 4.20: Árbol de distribución con router RP estático

Una vez realizada la prueba iperf se procede a enviar un flujo de vídeo y capturar los paquetes obteniendo los mismos paquetes que anteriormente, como se muestran en las figuras 4.21 y 4.22, por lo que este mecanismo también funciona.

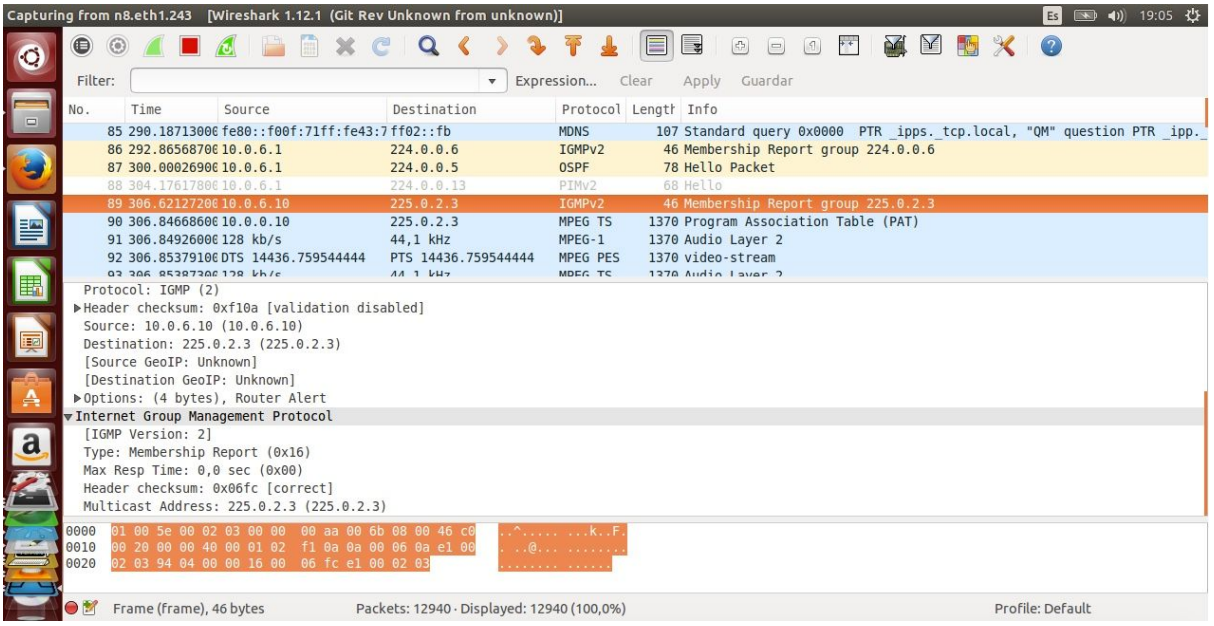

Figura 4.21: Primeros paquetes capturados en el router DR

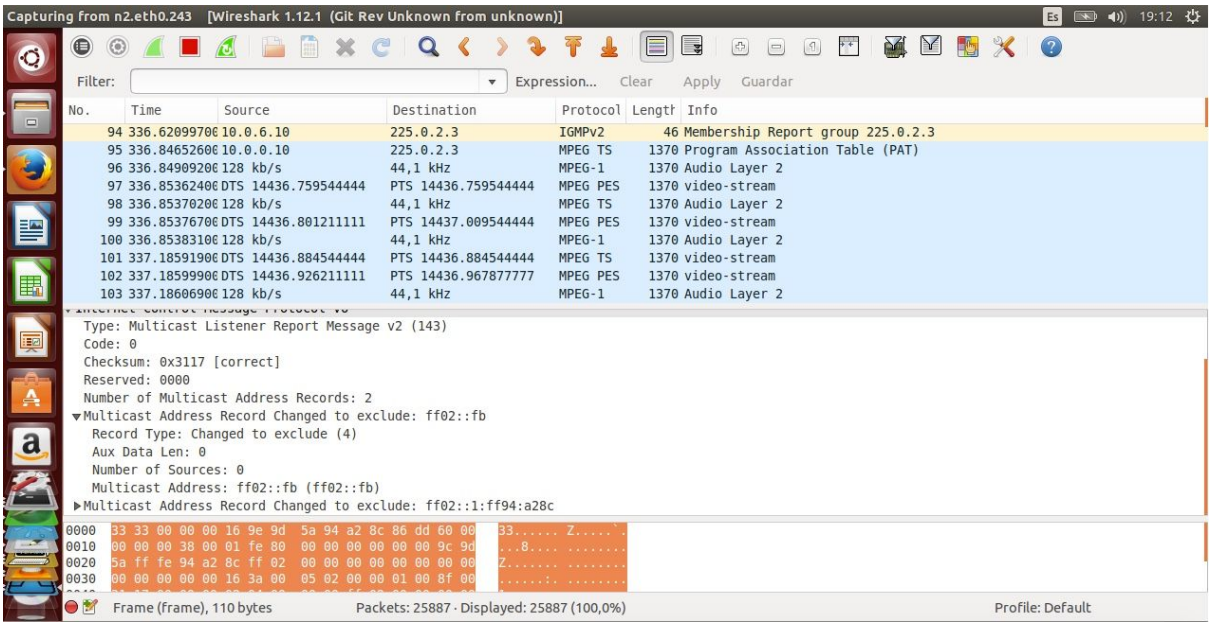

Figura 4.22: Primeros paquetes capturados en el host cliente

4.4. Evaluación de los resultados obtenidos

Como resultado de la emulación de los distintos escenarios presentados en este capítulo, se ha aprendido que los protocolos de encaminamiento funcionan correctamente y el tráfico de vídeo fluye por la red. A pesar del consumo en ancho de banda que implica la transmisión unicast, se ha podido observar que la recepción del flujo de vídeo es bastante buena. En cuanto al tráfico multicast se puede concluir que la replicación de paquetes en los routers puede ser un proceso costoso que conlleva a pérdida de paquetes.

Capítulo 5

ENTORNO SOCIO-ECONÓMICO

5.1. Planificación

Para la realización de este TFG se ha realizado una planificación en la que se ha estimado la duración de cada tarea destinada a la consecución de los objetivos, como se muestra en la figura 5.1.

ID	Tareas	Duración
1	**Investigación bibliografía**	20d
2	IPTV	12d
3	Protocolos de enrutamiento	8d
4	**Diseño arquitectura de red**	4d
5	Topología de red unicast	2d
6	Topologia de red multicast	2d
7	**Configuración servicios en CORE**	30d
8	Servicio RIP	7d
9	Servicio OSPF	7d
10	Servicio PIM-SM	16d
11	**Aplicación distribución de vídeo**	1d
12	**Emulaciones y pruebas**	15d
13	**Memoria**	30d

Figura 5.1: Planificación para realización de TFG

5.2. Presupuesto

A continuación se detalla un presupuesto para el presente TFG desglosando tanto el coste del personal como de los costes directos e indirectos.

5.2.1. Costes del personal

En este TFG se cuenta con un personal compuesto por dos personas, que son el estudiante y el tutor con roles de Ingeniero Junior e Ingeniero Senior respectivamente, donde sus retribuciones y el coste total se pueden observar en la figura 5.2.

	Horas por día	Coste por hora	Días	Coste
Estudiante	8	20€	100	16.000€
Tutor	2	40€	30	2.400€
			TOTAL	18.400€

Figura 5.2: Costes del personal

5.2.2. Costes de los recursos utilizados

Para la realización del proyecto se han necesitado ciertos recursos hardware y software como un ordenador portátil, la máquina virtual VirtualBox, el emulador de redes CORE y las plataformas que implementan protocolos de enrutamiento. En la figura 5.3 se muestran los recursos y su coste.

Descripción	Coste	Uso dedicado %	Dedicacion (meses)	Período de depreciació	Coste imputable
Ordenador portátil	990€	100	3	60	40€
VIrtualBox	0€	100	3	60	0€
CORE	0€	100	3	60	0€
Quagga	0€	50	3	60	0€
XORP	0€	50	3	60	0€
				TOTAL	40€

Figura 5.3: Coste de recursos

5.2.3. Otros costes directos

En este apartado se incluyen otros gastos no considerados anteriormente como los costes fungibles y los costes de transporte, como se muestra en las figuras 5.4 y 5.5. En los costes de transporte se ha considerado los 30 días que se ha reunido el personal utilizando un coche privado para el desplazamiento con una distancia de 15 km.

Descripción	Cantidad	Coste unitario	Coste imputable
Memoria USB	1	20€	20€
Material oficina	1	15€	15€
		TOTAL	35€

Figura 5.4: Costes fungibles

Precio por km	km	Coste imputable
0,17€	450	76,50€

Figura 5.5: Coste de transporte

5.2.4. Costes indirectos

Aquí se incluyen otros gastos que aunque no están relacionados con el Proyecto son necesarios para que se desarrolle, como la luz, el agua o la limpieza. Se realiza una estimación de estos costes del 20% sobre el resto de costes.

El resumen de costes se muestra en la figura 5.6, y el presupuesto final en la figura 5.7.

Concepto	Cantidad
Personal	18.400€
Amortización	40€
Coste de funcionamiento	111,50€
Costes indirectos	3.710,30€
TOTAL	22.262€

Figura 5.6: Resumen de costes

Concepto	Cantidad
Coste total	22.262€
Riesgo (20%)	4.452€
Beneficio (20%)	4.452€
TOTAL sin IVA	31.166€
IVA	6.545€
TOTAL	37.711€

Figura 5.7: Presupuesto final

Añadiendo al total de costes los márgenes por riesgo y beneficio, así como el IVA se obtiene que el presupuesto total de este proyecto es de **TREINTA Y SIETE MIL SETECIENTOS ONCE** euros.

5.3 Impacto socio-económico

Con el resultado obtenido de la realización de este proyecto, que se basa en crear un entorno de pruebas para tráfico de vídeo, se puede concluir que puede ser útil para experimentar y probar nuevas funcionalidades y mejoras en la red, como hacen actualmente proveedores de servicios. Por tanto, se obtiene un impacto económico positivo, al reducir los costes que implicaría hacer las mismas pruebas en equipos reales.

Capítulo 6

CONCLUSIONES

En este TFG se ha creado una plataforma de pruebas basada en un emulador de redes con el fin de proporcionar una herramienta que permita evaluar el tráfico de red de un servicio IPTV. Esta plataforma tiene diversas topologías de red configuradas con protocolos de encaminamiento unicast y multicast, que permite realizar pruebas de rendimiento de red.

El emulador de redes CORE es una herramienta muy útil, si bien es verdad que el manejo de sus funcionalidades conllevó muchas consultas al manual de usuario y ejecuciones de los comandos en repetidas ocasiones.

En cuanto a los protocolos de encaminamiento, su configuración no es tan complicada como al principio se podría esperar gracias a Internet y al manual de XORP. Se ha podido observar el correcto funcionamiento de la configuración de las tablas de enrutamiento para que los paquetes lleguen a su destino.

Se concluye que el resultado obtenido tras la realización de este proyecto es satisfactorio, pues se han conseguido realizar los objetivos, estudiar el sistema IPTV y protocolos de encaminamiento.

En cuanto a una posible mejora podría ser la implementación de una topología de red más realista basada en las que usan los servicios de proveedores de IPTV y de la que no se ha encontrado referencia en Internet.

Anexo A: Configuración de VLC

Cuando se lanza VLC en el nodo emulado aparece una interfaz gráfica como la que se muestra en la figura A.1.

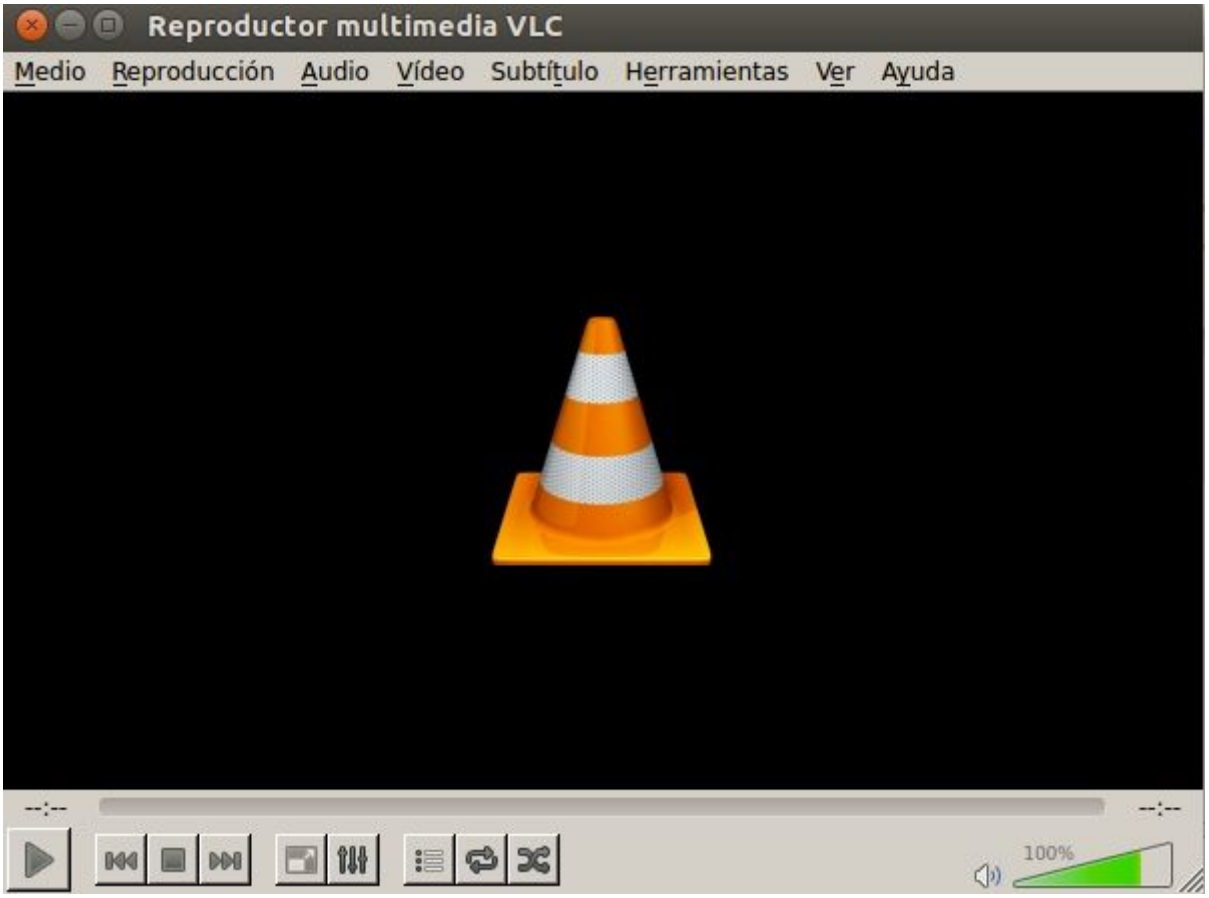

Figura A.1: Interfaz gráfica VLC en el nodo emulado

Para configurar la emisión de un vídeo se deben realizar los siguientes pasos:

- Paso 1: pulsar sobre el botón Medio de la barra de herramientas y seleccionar la opción Emitir del menú desplegable, como se muestra en las figuras A.2 y A.3.

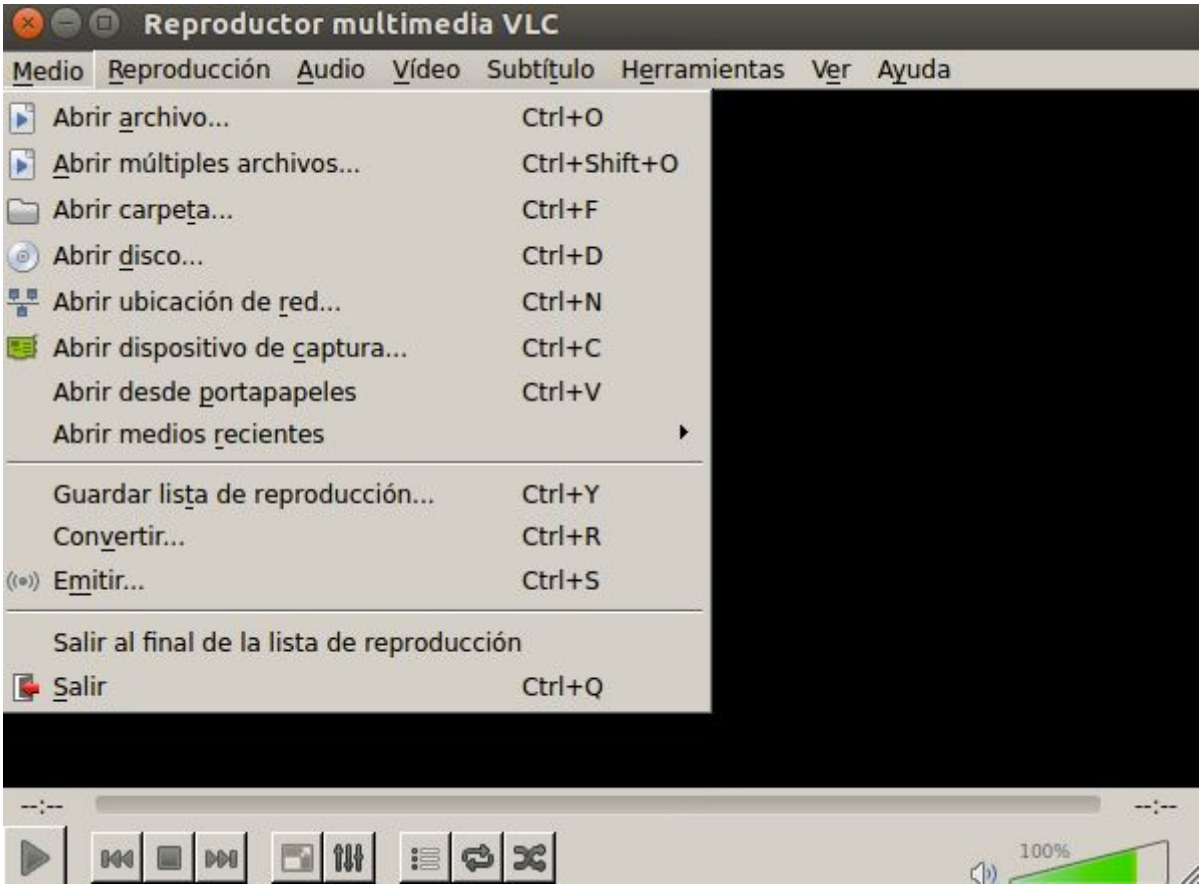

Figura A.2: Emitir

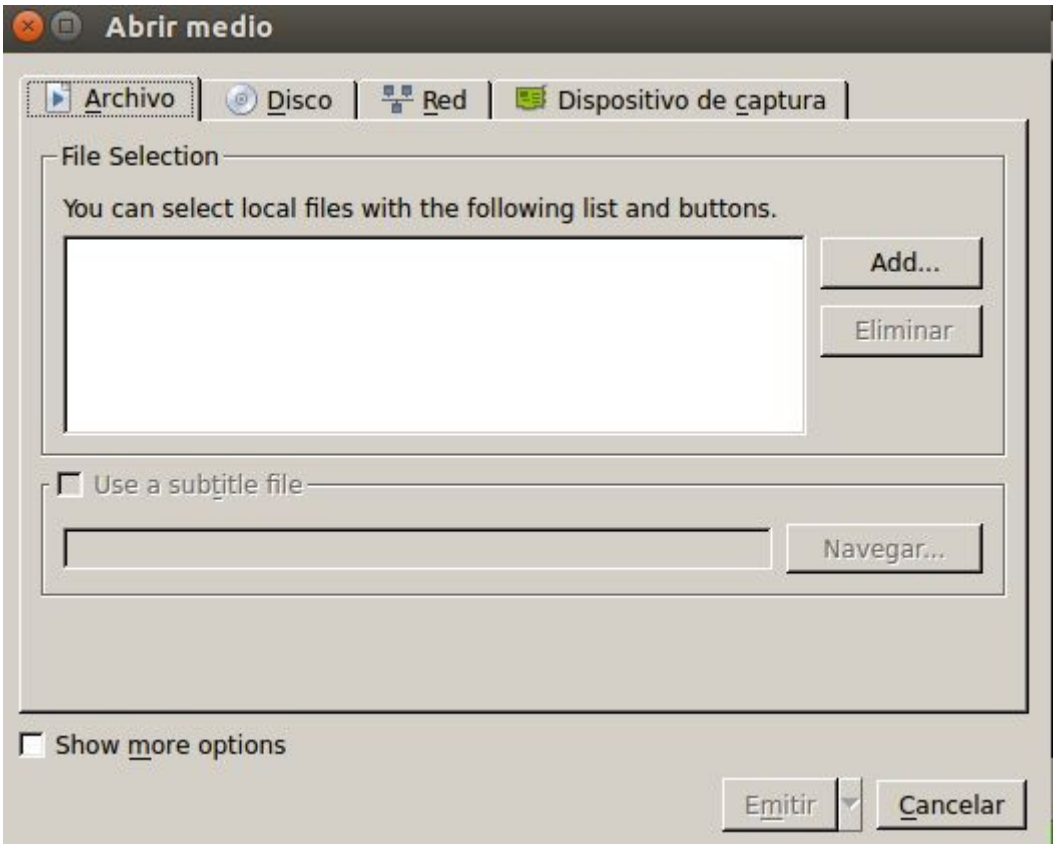

Figura A.3: Abrir medio

- Paso 2: pulsar sobre el botón Add y seleccionar el archivo que se quiere transmitir. Una vez elegido el archivo pulsar el botón Emitir.
- Paso 3: en la pantalla que aparece, que es la salida de emisión con la ubicación del archivo, pulsar Next.
- Paso 4: en esta pantalla elegir el protocolo que será utilizado, en nuestro caso RTP/MPEG Transport Stream, del menú desplegable, y pulsar Añadir. Esto se muestra en la figura A.4.

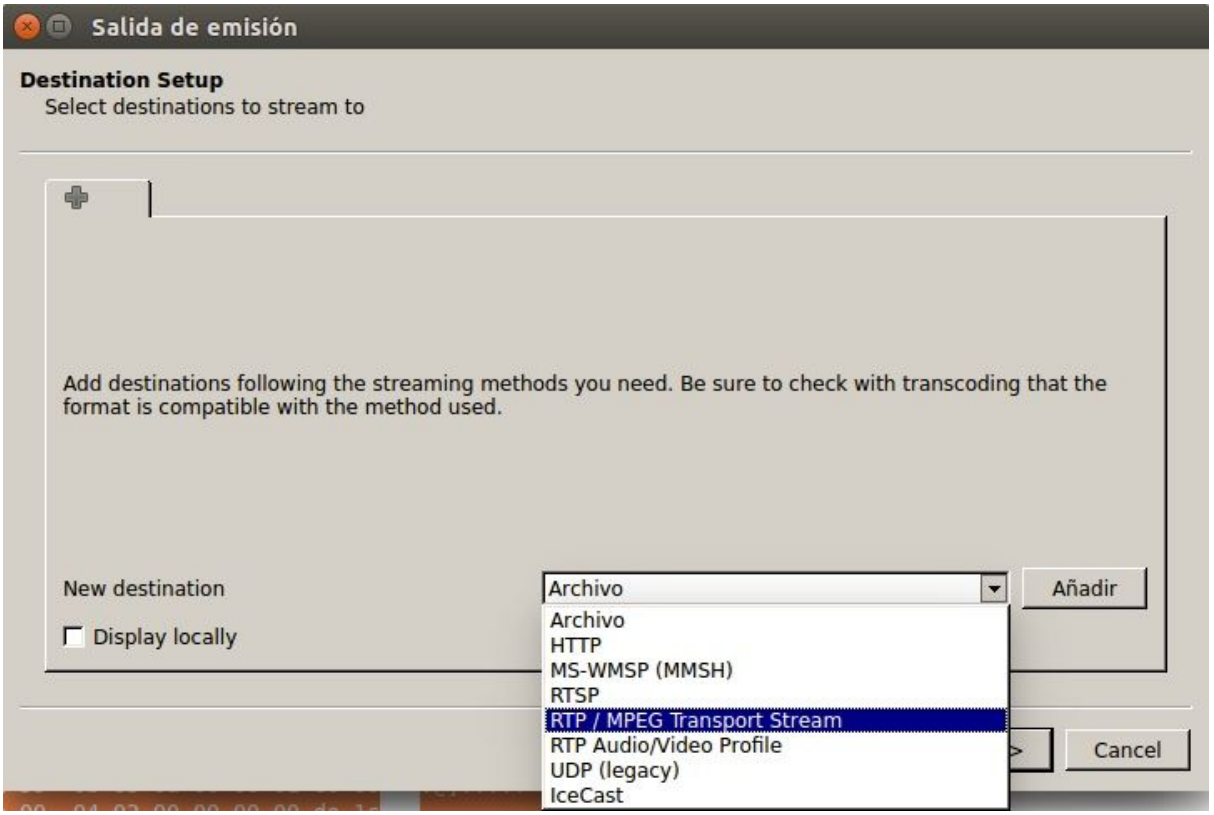

Figura A.4: Protocolo

- Paso 5: en esta pantalla se debe introducir la dirección IP y el puerto, que se puede dejar el que aparece por defecto, y pulsar el botón Next Esto se muestra en la figura A.5.

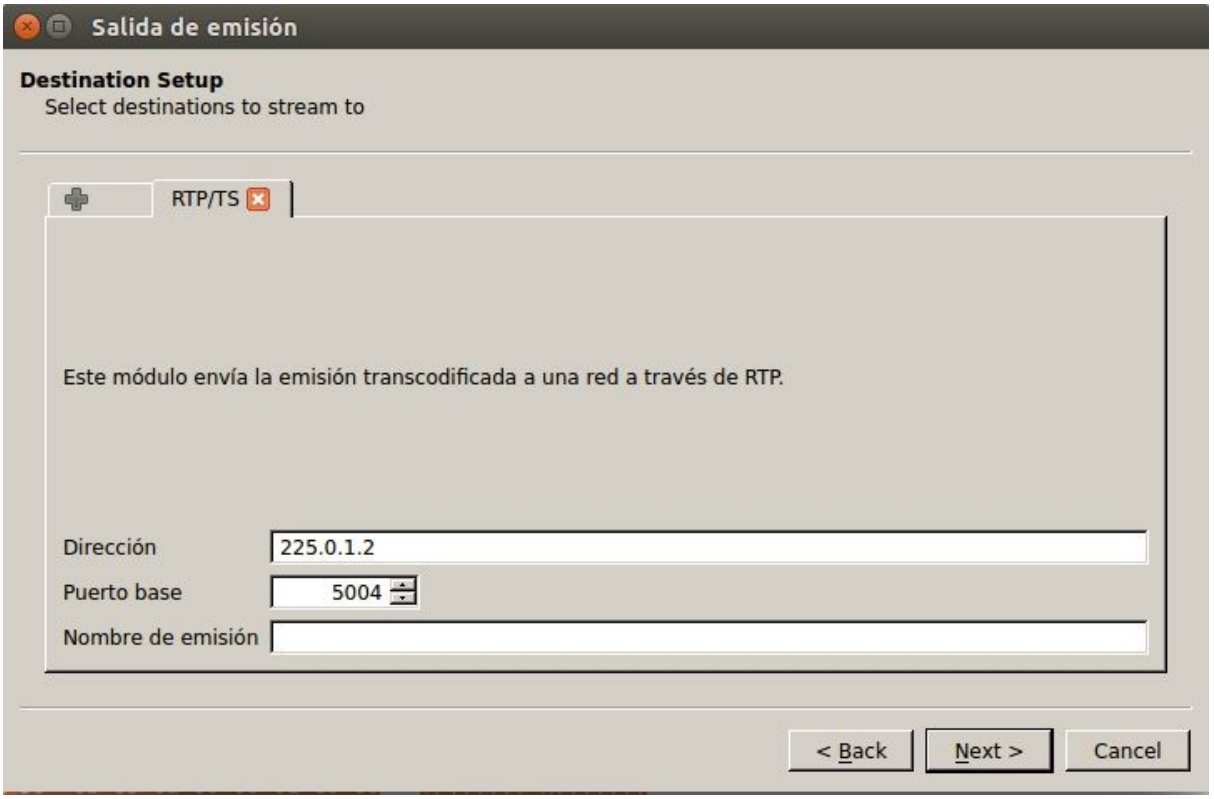

Figura A.5: Configuración dirección y puerto

- Paso 6: elegir el formato de codificación de la lista desplegable y activar o desactivar la función de transcodificación según interese. A continuación pulsar Next.
- Paso 7: Marcar la opción “Stream all elementary streams”, y en caso de tratarse de transmisión multicast se debe indicar el TTL en el string de salida de emisión generado como se muestra en la figura A.6, finalmente pulsar emitir, y a continuación comienza la emisión.

Para recibir la emisión se deben seguir los siguientes pasos:

- Paso 1: pulsar sobre el botón Medio y seleccionar la opción Abrir ubicación de red del menú desplegable.
- Paso 2: en la pantalla que se nos muestra introducir la URL de la red, en nuestro caso rtp://@225.0.1.2:5004 si se trata de multicast, o simplemente rtp://@5004 si se trata de unicast. A continuación pulsar el botón Reproducir para comenzar la reproducción. Esto se muestra en la figura A.7.

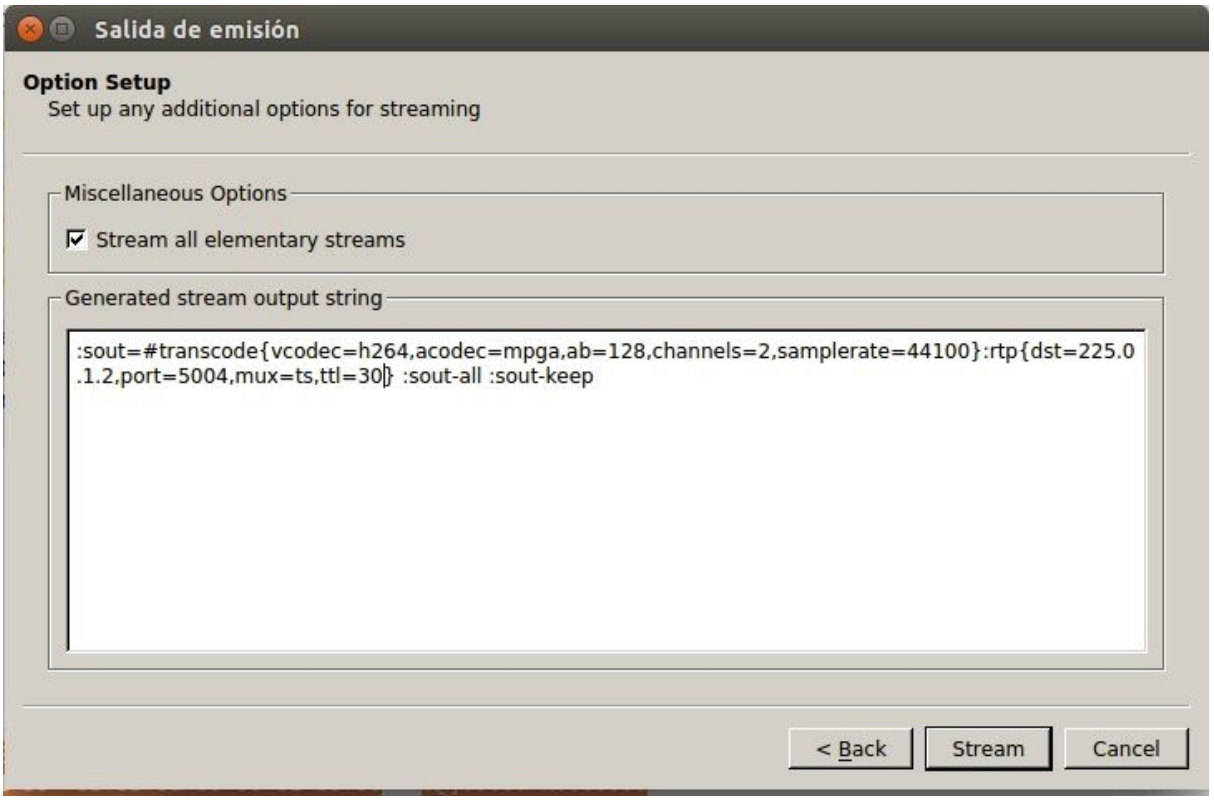

Figura A.6: Configuración TTL

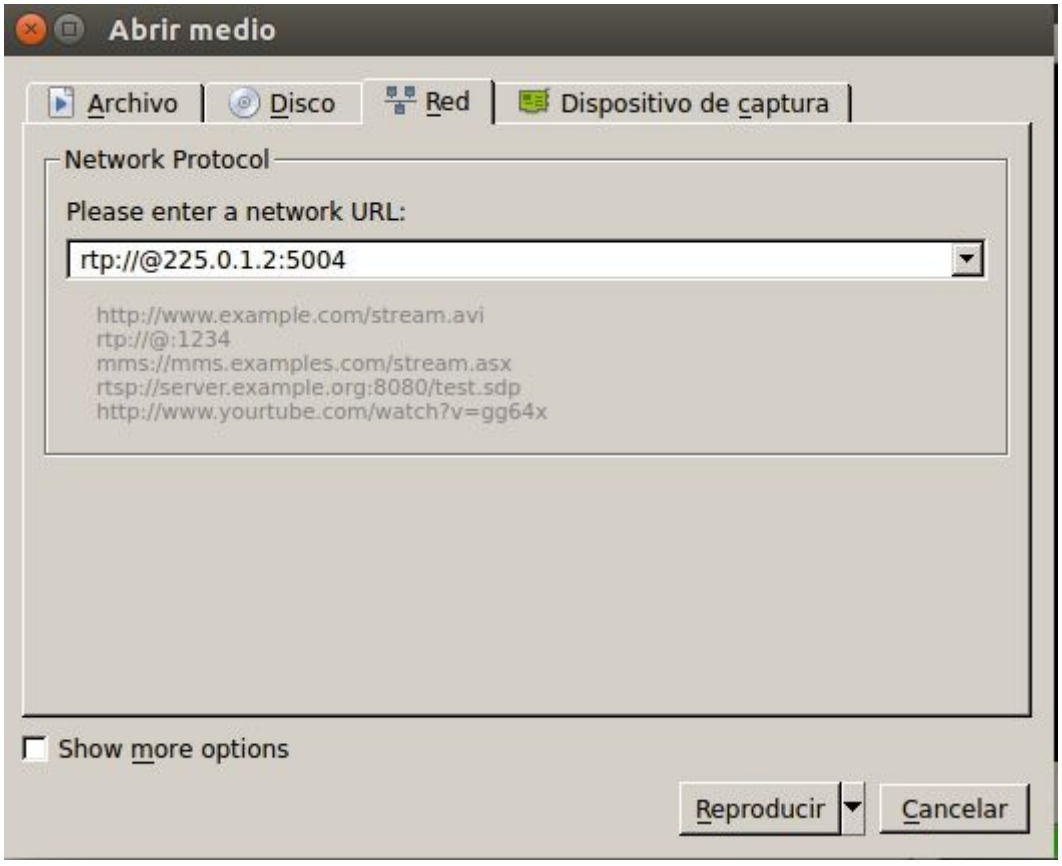

Figura A.7: Configuración recepción

Anexo B: Extended abstract

IPTV means television over IP protocol, it is not a protocol in itself, but a system of distribution of video signals over a broadband network. This service is based on video streaming so that full download is not necessary to start playback. This service differs from traditional television in that the video streams are sent when the user requests them. IPTV offers other services such as recording, pausing and start playback, as well as custom content on request. Delivering the service with good quality requires certain requirements such as high bandwidth, high signal-to-noise ratio and low attenuation. The content in standard definition, SDTV, generates a traffic of 3 Mbps, whereas in high definition, HDTV, it is 10 Mbps.

ETSI TISPAN defines the quality of experience (QoE) for multimedia services as "the user's experience when making use of communication services or applications provided by the CSP (Communication Service Provider), describing how the service and if it meets your expectations". Service providers must ensure a quality of experience in order to maintain commercial success.

Nowadays multimedia services consume a large amount of the network resources so that service providers are faced with the problem of dimensioning their networks in a way that can serve the demand. It is necessary to use testbed to evaluate the design of the network, to test changes and improvements. This reduces costs because it is not necessary to perform the same tests with real equipment.

The main objective of this TFG is to create a testing platform for multicast traffic based on a network emulator. For the implementation of the platform the work is divided into certain tasks, which are: design the network topology, configure the routing protocols in the CORE emulator, implement a distribution application, perform the different emulations for the delivery of video streams.

The network architecture of an IPTV system consists of a Super Head End (SHE), a Video Service Office (VSO) and a Local End Office (LEO). In the Super Head End,

the contents are received through satellite antennas or terrestrial networks, which are processed and converted to the coding format chosen to transmit them to the next installation. The Video Service Office receives content from the Super Head End and from local sources for processing and delivery to a geographic region. In the Local End Office, packet switching is performed and content is sent to each subscriber through the broadband access network. In the client's home a decoder called Set Top Box (STB) is needed that decodes the signal and has software to show the programming guide (EPG) and other graphics.

To deliver the IPTV service to households, DSL (Digital Subscriber Line) or FTTH (Fiber to the Home) technology can be used in the access network. Both technologies meet the bandwidth requirements, although fiber optics offers better results in terms of speed and electromagnetic interference.

IPTV systems use two transmission modes: unicast for VoD (Video on Demand) content and multicast for channels that are viewed simultaneously by different clients. The unicast mode is based on a one-to-one delivery where the sender sends a copy to each receiver. In multicast mode the one-to-many model is followed, the network being responsible for replicating the packets to get them to the different destinations.

Examples of unicast routing protocols are RIP (Routing Information Protocol), OSPF (Open Shortest Path First) and BGP (Border Gateway Protocol). While multicast routing protocols would be DVMRP (Distance Vector Multicast Routing Protocol), PIM-DM (Protocol Independent Multicast-Dense Mode) and PIM-SM (Protocol Independent Multicast-Sparse Mode) protocols.

RIP is an Interior Gateway Protocol (IGP) that uses the vector-distance routing algorithm to calculate the best metric to the destination. Routers exchange their routing tables with their neighboring routers, and if a router discovers that a metric has changed, broadcast it to the other routers.

OSPF is also an IGP protocol that uses the Dijkstra link-state algorithm and Link State Advertisement (LSA) to describe the topology and routing information. This protocol divides the domain into different areas obtaining scalability properties, and within each area all routers must have the same Link State Advertisement.

Multicast routing protocols are divided into dense and sparse protocols depending on the distribution of receivers interested in receiving packets from a certain multicast group. Dense protocols follow the flood and pruning model, so messages are sent to all routers reporting multicast sources. Routers that are not interested in receiving multicast traffic send a prune message, so the distribution tree is easily created.

Sparse protocols follow an explicit join model and instead of announcing the sources a router is determined, which will be the root of the distribution tree, and knows all the multicast sources.

DVMRP is a dense protocol that uses the vector-distance algorithm and can run on routers that are not enabled for multicast traffic. The router has a routing table with the multicast groups and the corresponding distances between the router and the destination.

PIM-DM is very similar in its operation to DVMRP, except that it uses a unicast protocol to populate the routing tables.

PIM-SM creates shared trees in which the root is a router called Rendezvous-Point (RP) to send multicast traffic to interested receivers. This router can be determined in three modes: RP-static, auto-RP and PIM Bootstrap. The shared tree may not be the best-metric option so receivers when they know the source can join their distribution tree.

The static RP router mechanism is based on a manual configuration of the chosen router's address as the rendezvous-point. The auto-RP mechanism is based on the candidates sending a RP advertisement message to address 224.0.1.39, and the routers configured as mapping agents decide the RP router according to some criteria. In PIM bootstrap some routers are configured as BSR candidates with a priority value; these routers send messages to all routers until only one BSR candidate remains; the candidate RP routers send messages to this BSR candidate; it adds the information in the bootstrap messages that reach the other routers; the routers perform a hash algorithm to determine the RP router.

For the realization of the project we have the CORE network emulator in which a network topology is designed for unicast and multicast traffic. You must enable unicast and multicast routing protocols, and use a video distribution and playback application to check network operation.

In CORE we have the concept of service to specify which processes are executed in the nodes implemented in the network. These services must be configured by creating the configuration files and indicating the directories in which these files are located.

The services we configure for this project are RIP for unicast traffic, and OSPF and PIM-SM for multicast traffic.

Configuration files are a set of commands and directives that must be accompanied by values, mostly IP addresses or integers. They serve to indicate the enabled interfaces that must be announced to create the routing tables, or to configure the Rendezvous-Point determination mechanism in the case of the PIM-SM protocol.

The CORE network emulator allows you to connect to real physical networks through the RJ45 and GRE tunnel nodes. This project uses the RJ45 node together with a dummy interface to connect to the host machine. It can also be connected to the host machine by activating the primary network control.

The video distribution application is based on the VLC media player and framework. In this way, VLC is launched in the nodes of the network scenario using SSH (Secure SHell) and the framework is configured to send and receive a video stream.

Configured the services in each of the scenarios we proceed to perform different tests to validate the platform implemented. The first of the tests is to check the connectivity between different nodes of the network, mainly between the host server and the client host. This is done through the programs ping and iperf that check the state of the links and the performance of the network respectively. Once the connectivity is verified in the emulated network scenarios, a video stream is delivered. Video delivery succeeds in all scenarios although image quality is not as good as it should be.

Different case studies are also performed in the multicast traffic network scenario to validate the implemented test environment. One is to display the data plane by capturing packets that are exchanged between the client host and the directly connected router.

Service providers offer their subscribers the ability to pause playback or return to program start, which means switching from multicast mode delivery to unicast mode with custom content. This is the other case study that is done in the project.

As a result of the emulation of the different scenarios configured in this project, it has been learned that the routing protocols work correctly and video traffic flows over the network.

It is concluded that the main objective of creating a test environment for multicast traffic has been achieved. This implemented platform could be used to test physical network topologies in order to design and evaluate them for implementation and improvement. Using emulated or simulated testing environments instead of real equipment offers cost savings.

It can be concluded that the development platform can be useful to experiment and test new functionalities and improvements in the network, as they do service providers.

As future implementation based on this project we can find a network topology that is currently used by service providers

Bibliografía

[JGR] Jean-Gabriel Remy, Charlotte Letamendia, “Home Area Networks and IPTV”, 2011, 1ª edición, Ed. ISTE

[Subs]http://www.panoramaaudiovisual.com/2011/09/30/espana-contara-con-44-millones-de-suscriptores-a-iptv-en-2020/, 5 Septiembre 2017

[VNI]https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.html, 5 Septiembre 2017

[WS07] Wes Simpson, Howard Greenfield, “IPTV and Internet Video: New Markets in Television Broadcasting”, 2007, 1ª edición, Ed. Focal Press

[HG09] Wes Simpson, Howard Greenfield, “IPTV and Internet Video: Expanding the Reach of Television Broadcasting”, 2009, 2ª edición, Ed. Focal Press

[GO08] Gerard O´Driscoll, “Next Generation IPTV Services and Technologies”, 2008, Ed. Ringgold Inc

[DR08] David H. Ramirez, “IPTV Security: Protecting High-Value Digital Contents”, 2008, Ed. Ringgold Inc

[CISCO]https://www.cisco.com/c/en/us/td/docs/ios/solutions_docs/ip_multicast/White_papers/mcst_ovr.html#wp1009068, 28 Jiliio 2017

[BE02] Brian M. Edwards, Leonard A. Giuliano, Brian R. Wright, “Interdomain Multicast Routing: Practical Juniper Networks and Cisco System Solutions”, 2002, Ed. Addison-Wesley

[RIPv1] C. Hedrick, Routing Information Protocol, IETF RFC 1058, Junio 1988

[RIPv2] G. Malkin, Routing Information Protocol Version 2, IETF RFC 2453, Noviembre 1998

[RIPng] G. Malkin, R. Minnear, RIPng for IPv6, IETF RFC 2080, Enero 2007

[OSPFv2] J. Moy, OSPF Version 2, IETF RFC 2328, Abril 1998

[DVMRP] T. Pusateri, Distance Vector Multicast Routing Protocol, IETF Internet draft-ietf-imdr-dvmrp-v3-11, Octubre 2003

[PIMDM] A. Adams, J. Nicholas, W. Siadak, Protocol Independent Multicast-Dense Mode (PIM-DM): Protocol Specification (Revised), IETF RFC 3973, Enero 2006

[PIMSM] B. Fenner, M. Handley, H. Holbrook, I. Kouvelas, R. Parekh, Z. Zhang, L. Zheng, Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification (Revised), IETF RFC 7761, Agosto 2006

[CORE] https://www.nrl.navy.mil/itd/ncs/products/core

[QGG] http://www.nongnu.org/quagga/

[XORP] http://www.xorp.org/

[SMRT]https://books.google.es/books?id=jIXQOMEBC4kC&dq=judith+estrin+bill+carrico+iptv&hl=es&source=gbs_navlinks_s, 4 Septiembre 2017

[UIT] http://www.itu.int/en/ITU-T/focusgroups/iptv/Pages/default.aspx, 8 Septiembre 2017

[DVB] https://www.dvb.org/standards, 12 Septiembre 2017

[IPTV]https://www.cisco.com/c/es_mx/solutions/service-provider/iptv-solutions-wireline-carriers/index.html, 13 Agosto 2017

[IGMPv1] S. Deering, Host Extensions for IP Multicasting, IETF RFC 1112, Agosto 1989

[IGMPv2] W. Fenner, Internet Group Management Protocol, Version 2, IETF RFC 2236, Noviembre 1997

[IGMPv3] B. Cain, S. Deering, I. Kouvelas, B. Fenner, A. Thyagarajan, Internet Group Management Protocol, Version 3, IETF RFC 3376, Octubre 2002